

**PENGEMBANGAN SISTEM INFORMASI *SALES FORCE*
AUTOMATION (SFA) MENGGUNAKAN METODE
PROTOTYPING PADA BRANDED IT STORE MALANG**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Ageng Rama Wijaya
NIM: 135150407111050



PROGRAM STUDI SISTEM INFORMASI
JURUSAN SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018

PENGESAHAN

PENGEMBANGAN SISTEM INFORMASI SALES FORCE AUTOMATION(SFA)
MENGUNAKAN METODE PROTOTYPING PADA BRANDED IT STORE MALANG

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Ageng Rama Wijaya
NIM: 135150407111050

Skripsi ini telah diuji dan dinyatakan lulus pada
3 Mei 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I



Yusi Tyroni Mursityo, S.Kom., M.AB
NIP. 19800228 200604 1 001

Dosen Pembimbing II

Ka - PS - SI



Niken Hendrakusma Wardani, S.Kom., M.Kom.
NIK. 201606 900621 2 001

Mengetahui

Ketua Jurusan Sistem Informasi



Dr. Eng., Herman Tolle, S.T, M.T.
NIP: 19740823 200012 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 3 Mei 2018



Ageng Rama Wijaya

NIM: 135150407111050

KATA PENGANTAR

Segala puji syukur bagi Allah SWT karena berkat rahmat dan karunia yang telah diberikan penulis dapat menyelesaikan penyusunan skripsi yang berjudul “Pengembangan Sistem Informasi Sales Force Automation(SFA) Menggunakan Metode Prototyping Pada Branded IT Store Malang”. Shalawat serta salam senantiasa penulis curahkan kepada Nabi Muhammad SAW, kepada keluarganya dan para sahabatnya.

Dalam penyusunan dan penulisan naskah skripsi ini penulis mendapatkan banyak bantuan, bimbingan serta dukungan dari berbagai pihak baik secara moril maupun materil. Dalam kesempatan ini penulis ingin mengucapkan banyak terima kasih kepada yang telah membantu dalam penyelesaian skripsi ini diantaranya:

1. Kedua orang tua penulis, Ibu Sunarsih dan Bapak Dody Murtiwidigdo yang telah memberikan semua kemampuan mereka dalam memberikan motivasi, nasihat serta doa yang tiada hentinya kepada penulis dalam pengerjaan skripsi ini.
2. Bapak Yusi Tyroni Mursityo, S.Kom., M.Ab, selaku pembimbing satu yang telah memberikan waktu, tenaga serta mengarahkan ketegasan dan membuka wawasan dalam pelaksanaan bimbingan.
3. Ibu Niken Hendrakusma Wardani, S.Kom., M.Kom., selaku pembimbing dua yang telah sabar memberikan waktu, tenaga serta motivasi dalam pelaksanaan bimbingan.
4. Ibu Merry Oen selaku *Manager* sekaligus penanggung jawab Branded IT Store Malang yang telah memberi kesempatan untuk dan kemudahan dalam melakukan penelitian di toko sehingga sangat membantu dalam penyelesaian skripsi ini.
5. Teman-teman dekat sistem informasi yang tidak dapat penulis sebutkan namanya satu per satu, yang telah memberikan penulis dukungan moril, motivasi serta saran dalam pengerjaan skripsi ini.
6. Keluarga Besar Sistem Informasi yang telah menerima penulis dengan baik yang tidak bisa penulis sebutkan namanya satu per satu.
7. Karyawan dan staf Fakultas Ilmu Komputer Universitas Brawijaya yang telah membantu dan memudahkan dalam segala pengurusan pada skripsi ini.
8. Serta seluruh pihak yang telah membantu dan memudahkan dalam kelancaran penulisan skripsi yang tidak bisa penulis sebutkan satu per satu.

Dalam penulisan skripsi ini tentunya tidak lepas dari kekurangan, baik kualitas maupun kuantitas dari materi yang disajikan. Penulis menyadari bahwa sripsi ini masih jauh dari kata sempurna, semua ini didasarkan dari keterbatasan yang dimiliki penulis.

Malang, 3 Mei 2018

Penulis

agengramawijaya@gmail.com



ABSTRAK

Branded IT Store merupakan salah satu perusahaan yang bergerak pada bidang penjualan barang elektronik. Penjualan yang terjadi hampir setiap hari, membutuhkan pengelolaan data yang baik. Sedangkan di Branded IT Store masih menggunakan cara manual untuk melakukan pencatatan transaksi penjualan. Perusahaan masih belum mempunyai sistem yang menghubungkan data terkait dengan produk atau penjualan yang dimiliki oleh perusahaan.

Sales Force Automation(SFA) adalah sebuah sistem yang dapat membantu perusahaan untuk meningkatkan penjualan dengan memberikan kemudahan berupa otomatisasi pada bidang penjualan berdasarkan data yang ada. *Sales force automation* dapat memberikan solusi untuk mengembangkan sebuah sistem yang mempunyai kegunaan untuk menyelesaikan masalah pencatatan transaksi penjualan. Beberapa fungsi dari *sales force automation* digunakan untuk mempermudah proses transaksi penjualan, pengelolaan data produk, pengelolaan data pelanggan serta pengelolaan data pegawai. Pengembangan pada penelitian ini menggunakan metode *prototyping* untuk memudahkan memahami kebutuhan pengguna pada tahap awal. Pengembangan dimulai dengan melakukan analisis kebutuhan untuk mendapatkan gambaran kebutuhan sistem dari pengguna, setelah itu membuat perancangan sistem yang akan digunakan untuk melakukan implementasi pada sistem. Hasil dari implementasi sistem akan diuji untuk menentukan apakah terdapat masalah dalam sistem, sehingga dapat teridentifikasi jika sistem berjalan sesuai dengan persyaratan atau tidak.

Hasil pengujian pada sistem informasi *sales force automation* menghasilkan status valid pada pengujian validasi pada fungsi pengelolaan data produk, pengelolaan data penjualan, pengelolaan data pegawai, pengelolaan data pelanggan, dan melakukan pembelian dan dapat berjalan dengan baik. Berdasarkan hasil UAT yang didapatkan, menunjukkan bahwa sistem dapat diterima 100% dari sisi perusahaan dimana tester adalah manager, penjual dan kasir. Pada hasil pengujian kompatibilitas terdapat masalah minor pada aplikasi web browser Internet Explorer 9 sampai dengan 11, web browser Firefox versi 58 dan aplikasi web browser BlackBerry dengan versi 7.1 atau sebelumnya.

Kata kunci: *Sales Force Automation*, pelanggan, web, prtotyping

ABSTRACT

IT branded Store is one of the companies engaged in the sale of electronic goods. The sale happened almost every day, require a good data management. Whereas in Branded IT Store still use the manual way to do a recording of a sale. The company still has not had a system that connects the data related to the product or sales that are owned by the company.

Sales Force Automation (SFA) is a system that can help the company to increase sales by providing convenience in the form of automation in the field of sales based on existing data. Sales force automation can provide solutions to develop a system that has uses to resolve the problem of the recording of a sale. Some of the functions of sales force automation is used to simplify the transaction process, product data management, sales management, customer data as well as data management employees. Development on this research is using prototyping method, it make easier to understand users' needs at an early stage. The development began by conducting a needs analysis to get an overview of the system requirements of the user, after that the next stage is making the design of the system to be used for the implementation of the system. The result of the implementation of the system will be tested to determine if there is a problem in the system or not, so it can be identified if the system is running in accordance with the requirements or not.

The results of validation testing of this research on the functions of product data management, sales data management, employee data management, data management, and purchased by customer can run well and valid. UAT result shows that the system 100% acceptable from the company whose the tester are the manager, the seller and the cashier. On the compatibility testing results, there are minor problems on the application web browser Internet Explorer 9 to 11 with the Firefox web browser, version 58 and web applications BlackBerry browser with version 7.1 or earlier.

Keywords: Sales Force Automation, customer, web, prototyping

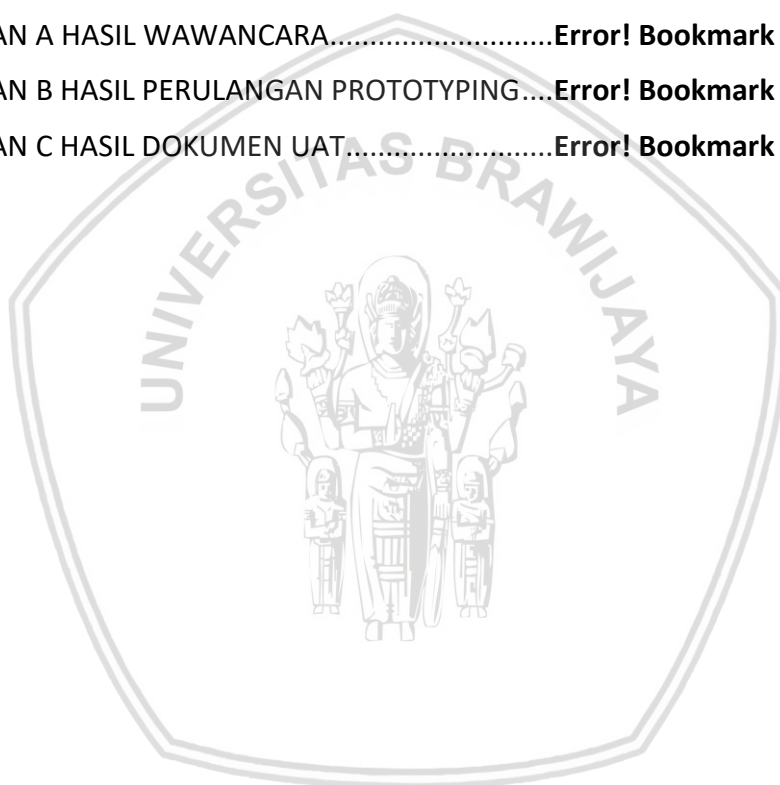
DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiv
DAFTAR LAMPIRAN	xvi
BAB 1 PENDAHULUAN.....	Error! Bookmark not defined.
1.1 Latar belakang.....	Error! Bookmark not defined.
1.2 Rumusan masalah.....	Error! Bookmark not defined.
1.3 Tujuan	Error! Bookmark not defined.
1.4 Manfaat.....	Error! Bookmark not defined.
1.5 Batasan masalah	Error! Bookmark not defined.
1.6 Sistematika pembahasan.....	Error! Bookmark not defined.
BAB 2 LANDASAN KEPUSTAKAAN	Error! Bookmark not defined.
2.1 Kajian Pustaka	Error! Bookmark not defined.
2.2 Profil Perusahaan	Error! Bookmark not defined.
2.3 <i>Customer Relationship Management</i>	Error! Bookmark not defined.
2.3.1 Pengertian <i>Customer Relationship Management</i>	Error! Bookmark not defined.
2.3.2 Tujuan dan Manfaat CRM	Error! Bookmark not defined.
2.3.3 Jenis-Jenis CRM	Error! Bookmark not defined.
2.3.4 Pengertian <i>Sales Force Automation (SFA)</i>	Error! Bookmark not defined.
2.4 Proses Bisnis.....	Error! Bookmark not defined.
2.5 <i>Business Process Model Notation</i>	Error! Bookmark not defined.
2.5.1 <i>Flow Object</i>	Error! Bookmark not defined.
2.5.2 <i>Connections</i>	Error! Bookmark not defined.

2.5.3 Swimlanes	Error! Bookmark not defined.
2.5.4 Artifacts	Error! Bookmark not defined.
2.6 Software Development Life Cycle (SDLC)	Error! Bookmark not defined.
2.6.1 Model Prototyping	Error! Bookmark not defined.
2.7 Unified Modeling Language (UML)	Error! Bookmark not defined.
2.7.1 Use Case Diagram	Error! Bookmark not defined.
2.7.2 Activity Diagram	Error! Bookmark not defined.
2.7.3 Sequence Diagram	Error! Bookmark not defined.
2.7.4 Class Diagram	Error! Bookmark not defined.
2.8 Physical Data Model	Error! Bookmark not defined.
2.9 CodeIgniter	Error! Bookmark not defined.
2.9.1 Model, View dan Controller pada CI	Error! Bookmark not defined.
2.10 Pengujian Perangkat Lunak	Error! Bookmark not defined.
2.10.1 Validation Testing	Error! Bookmark not defined.
2.10.2 User Acceptance Testing	Error! Bookmark not defined.
2.10.3 Compatibility Testing	Error! Bookmark not defined.
BAB 3 METODOLOGI	Error! Bookmark not defined.
3.1 Studi Literatur	Error! Bookmark not defined.
3.2 Pengumpulan Data	Error! Bookmark not defined.
3.3 Analisis Kebutuhan	Error! Bookmark not defined.
3.4 Perancangan	Error! Bookmark not defined.
3.5 Implementasi	Error! Bookmark not defined.
3.6 Evaluasi Pelanggan	Error! Bookmark not defined.
3.7 Pengujian	Error! Bookmark not defined.
3.8 Kesimpulan & Saran	Error! Bookmark not defined.
BAB 4 ANALISIS KEBUTUHAN	Error! Bookmark not defined.
4.1 Deskripsi Umum Perangkat Lunak	Error! Bookmark not defined.
4.2 Analisis Proses Bisnis	Error! Bookmark not defined.
4.2.1 Proses Bisnis As-Is	Error! Bookmark not defined.
4.2.2 Proses Bisnis To-Be	Error! Bookmark not defined.
4.3 Identifikasi Aktor	Error! Bookmark not defined.
4.4 Spesifikasi Kebutuhan Sistem	Error! Bookmark not defined.

4.4.1 Fitur Sistem	Error! Bookmark not defined.
4.4.2 Kebutuhan Fungsional.....	Error! Bookmark not defined.
4.4.3 Kebutuhan Non-Fungsional	Error! Bookmark not defined.
4.5 Pemodelan <i>Use- Case Diagram</i>	Error! Bookmark not defined.
4.5.1 Spesifikasi <i>Use Case</i>	Error! Bookmark not defined.
4.6 Pemodelan <i>Activity Diagram</i>	Error! Bookmark not defined.
BAB 5 PERANCANGAN.....	Error! Bookmark not defined.
5.1 Perancangan <i>Sequence Diagram</i>	Error! Bookmark not defined.
5.2 Perancangan <i>Class Diagram</i>	Error! Bookmark not defined.
5.3 Perancangan Basis Data.....	Error! Bookmark not defined.
5.4 Perancangan Komponen.....	Error! Bookmark not defined.
5.4.1 Algoritma Method tambah_penjualan	Error! Bookmark not defined.
5.4.2 Algoritma Method update_pegawai	Error! Bookmark not defined.
5.5 Perancangan Antarmuka	Error! Bookmark not defined.
5.5.1 Halaman Login.....	Error! Bookmark not defined.
5.5.2 Halaman Daftar Data Produk	Error! Bookmark not defined.
5.5.3 Halaman Tambah Data Produk	Error! Bookmark not defined.
5.5.4 Halaman Beranda Pelanggan	Error! Bookmark not defined.
BAB 6 IMPLEMENTASI	Error! Bookmark not defined.
6.1 Spesifikasi Sistem	Error! Bookmark not defined.
6.1.1 Spesifikasi Perangkat Keras.....	Error! Bookmark not defined.
6.1.2 Spesifikasi Perangkat Lunak	Error! Bookmark not defined.
6.2 Implementasi Algoritma	Error! Bookmark not defined.
6.2.1 Algoritma Mengelola Penjualan.....	Error! Bookmark not defined.
6.2.2 Algoritma Mengelola Pegawai	Error! Bookmark not defined.
6.3 Implementasi Antarmuka	Error! Bookmark not defined.
6.3.1 Halaman <i>Login</i>	Error! Bookmark not defined.
6.3.2 Halaman Kelola Data Produk	Error! Bookmark not defined.
6.3.3 Halaman Kelola Data Penjualan	Error! Bookmark not defined.
6.3.4 Halaman Beranda Pelanggan	Error! Bookmark not defined.
6.3.5 Halaman Rincian Belanja.....	Error! Bookmark not defined.

6.4 Evaluasi Pelanggan.....	Error! Bookmark not defined.
BAB 7 PENGUJIAN	Error! Bookmark not defined.
7.1 Validation Testing	Error! Bookmark not defined.
7.2 User Acceptance Testing	Error! Bookmark not defined.
7.3 Compatibility Testing	Error! Bookmark not defined.
BAB 8 PENUTUP	Error! Bookmark not defined.
8.1 Kesimpulan.....	Error! Bookmark not defined.
8.2 Saran	Error! Bookmark not defined.
DAFTAR PUSTAKA.....	Error! Bookmark not defined.
LAMPIRAN A HASIL WAWANCARA.....	Error! Bookmark not defined.
LAMPIRAN B HASIL PERULANGAN PROTOTYPING	Error! Bookmark not defined.
LAMPIRAN C HASIL DOKUMEN UAT.....	Error! Bookmark not defined.



DAFTAR TABEL

Tabel 2.1 Simbol-simbol pada tipe <i>event</i>	Error! Bookmark not defined.
Tabel 2.2 Simbol-simbol pada tipe <i>activity</i>	Error! Bookmark not defined.
Tabel 2.3 Simbol-simbol pada tipe <i>gateway</i>	Error! Bookmark not defined.
Tabel 2.4 Simbol-simbol pada tipe <i>connections</i>	Error! Bookmark not defined.
Tabel 2.5 Simbol-simbol pada tipe <i>swimlanes</i>	Error! Bookmark not defined.
Tabel 2.6 Simbol-simbol pada tipe <i>artifacts</i>	Error! Bookmark not defined.
Tabel 2.7 Simbol-Simbol Use Case Diagram	Error! Bookmark not defined.
Tabel 2.8 Simbol-Simbol <i>Activity Diagram</i>	Error! Bookmark not defined.
Tabel 2.9 Simbol-Simbol <i>Sequence Diagram</i>	Error! Bookmark not defined.
Tabel 4.1 Deskripsi umum perangkat lunak.....	Error! Bookmark not defined.
Tabel 4.2 Identifikasi Aktor	Error! Bookmark not defined.
Tabel 4.3 Fitur Sistem.....	Error! Bookmark not defined.
Tabel 4.4 Kebutuhan Fungsional.....	Error! Bookmark not defined.
Tabel 4.5 Kebutuhan Non-Fungsional.....	Error! Bookmark not defined.
Tabel 4.6 Pemetaan Proses Bisnis <i>To-Be</i> dengan <i>Use Case</i>	Error! Bookmark not defined.
Tabel 4.7 Spesifikasi <i>Use Case Login</i>	Error! Bookmark not defined.
Tabel 4.8 Spesifikasi <i>Use Case</i> Mengelola Pegawai ..	Error! Bookmark not defined.
Tabel 4.9 Spesifikasi <i>Use Case</i> Melihat Pencapaian Pegawai	Error! Bookmark not defined.
Tabel 4.10 Spesifikasi <i>Use Case</i> Melihat Laporan Penjualan	Error! Bookmark not defined.
Tabel 4.11 Spesifikasi <i>Use Case</i> Mengelola Data Produk	Error! Bookmark not defined.
Tabel 4.12 Spesifikasi <i>Use Case</i> Mengelola Data Pelanggan	Error! Bookmark not defined.
Tabel 4.13 Spesifikasi <i>Use Case</i> Mengelola Data Penjualan	Error! Bookmark not defined.
Tabel 4.14 Spesifikasi <i>Use Case</i> Melihat Produk.....	Error! Bookmark not defined.
Tabel 4.15 Spesifikasi <i>Use Case</i> Membeli Produk.....	Error! Bookmark not defined.
Tabel 5.1 Algoritma Fungsi tambah_penjualan	Error! Bookmark not defined.
Tabel 5.2 Algoritma Fungsi update_pegawai.....	Error! Bookmark not defined.

Tabel 6.1 Spesifikasi Perangkat Keras	Error! Bookmark not defined.
Tabel 6.2 Spesifikasi Perangkat Lunak	Error! Bookmark not defined.
Tabel 6.3 Fungsi tambah_penjualan	Error! Bookmark not defined.
Tabel 6.4 Fungsi update_pegawai.....	Error! Bookmark not defined.
Tabel 7.1 Menambah Data Produk	Error! Bookmark not defined.
Tabel 7.2 Menambah Data Produk Alternatif 1: Data Produk Yang Dimasukkan Belum Lengkap	Error! Bookmark not defined.
Tabel 7.3 Memperbarui Data Produk	Error! Bookmark not defined.
Tabel 7.4 Menambah Data Penjualan.....	Error! Bookmark not defined.
Tabel 7.5 Memperbarui Data Penjualan	Error! Bookmark not defined.
Tabel 7.6 Menampilkan Data Penjualan	Error! Bookmark not defined.
Tabel 7.7 Menambah Data Pelanggan	Error! Bookmark not defined.
Tabel 7.8 Menambah Data Pelanggan Alternatif 1: Data Pelanggan Yang Dimasukkan Sudah Ada.....	Error! Bookmark not defined.
Tabel 7.9 Memperbarui Data Pelanggan	Error! Bookmark not defined.
Tabel 7.10 Menambah Data Pegawai	Error! Bookmark not defined.
Tabel 7.11 Menghapus Data Pegawai.....	Error! Bookmark not defined.
Tabel 7.12 Melihat Barang	Error! Bookmark not defined.
Tabel 7.13 Membeli Barang	Error! Bookmark not defined.
Tabel 7.14 Stakeholder yang terlibat pada <i>User Acceptance Testing</i>	Error! Bookmark not defined.
Tabel 7.15 Kriteria UAT	Error! Bookmark not defined.
Tabel 7.16 Task yang Perlu Dikerjakan Tester	Error! Bookmark not defined.
Tabel 7.17 Pertanyaan UAT.....	Error! Bookmark not defined.
Tabel 7.18 Hasil Wawancara UAT	Error! Bookmark not defined.

DAFTAR GAMBAR

Gambar 2.1 Struktur Organisasi Branded IT Store Malang **Error! Bookmark not defined.**

Gambar 2.2 Prototyping sebagai alat untuk kebutuhan perangkat lunak **Error! Bookmark not defined.**

Gambar 2.3 Diagram Pengembangan Evolutionary Prototyping **Error! Bookmark not defined.**

Gambar 3.1 Diagram Alur Penelitian **Error! Bookmark not defined.**

Gambar 4.1 Proses Bisnis Pembelian dari Supplier As-Is **Error! Bookmark not defined.**

Gambar 4.2 Proses Bisnis Penjualan As-Is..... **Error! Bookmark not defined.**

Gambar 4.3 Proses Bisnis Pelunasan As-Is..... **Error! Bookmark not defined.**

Gambar 4.4 Proses Bisnis Pembelian dari Supplier To-Be **Error! Bookmark not defined.**

Gambar 4.5 Proses Bisnis Penjualan To-Be..... **Error! Bookmark not defined.**

Gambar 4.6 Proses Bisnis Pelunasan To-Be **Error! Bookmark not defined.**

Gambar 4.7 Kodifikasi Fitur **Error! Bookmark not defined.**

Gambar 4.8 Kodifikasi Kebutuhan Fungsional **Error! Bookmark not defined.**

Gambar 4.9 Use Case Diagram..... **Error! Bookmark not defined.**

Gambar 4.10 Activity Diagram Login **Error! Bookmark not defined.**

Gambar 4.11 Activity Diagram Mengelola Data Pegawai **Error! Bookmark not defined.**

Gambar 4.12 Activity Diagram Mengelola Data Penjualan **Error! Bookmark not defined.**

Gambar 4.13 Activity Diagram Mengelola Data Produk **Error! Bookmark not defined.**

Gambar 5.1 Sequence Diagram Login..... **Error! Bookmark not defined.**

Gambar 5.2 Sequence Diagram Tambah Penjualan . **Error! Bookmark not defined.**

Gambar 5.3 Sequence Diagram Perbarui Penjualan. **Error! Bookmark not defined.**

Gambar 5.4 Sequence Diagram Tambah Data Pegawai **Error! Bookmark not defined.**

Gambar 5.5 Sequence Diagram Hapus Data Pegawai **Error! Bookmark not defined.**

Gambar 5.6 Sequence Diagram Perbarui Pegawai ... **Error! Bookmark not defined.**

Gambar 5.7 Sequence Diagram Tambah Data Produk **Error! Bookmark not defined.**

Gambar 5.8 Sequence Diagram Perbarui Data Produk **Error! Bookmark not defined.**

Gambar 5.9 Class Diagram Controller **Error! Bookmark not defined.**

Gambar 5.10 Class Diagram Model **Error! Bookmark not defined.**

Gambar 5.11 Physical Data Model **Error! Bookmark not defined.**

Gambar 5.12 Rancangan Halaman Login **Error! Bookmark not defined.**

Gambar 5.13 Rancangan Halaman Daftar Data Produk **Error! Bookmark not defined.**

Gambar 5.14 Rancangan Halaman Tambah Data Produk **Error! Bookmark not defined.**

Gambar 5.15 Rancangan Halaman Beranda Pelanggan **Error! Bookmark not defined.**

Gambar 6.1 Hasil Implementasi Halaman Login Admin **Error! Bookmark not defined.**

Gambar 6.2 Hasil Implementasi Halaman Login Pelanggan **Error! Bookmark not defined.**

Gambar 6.3 Hasil Implementasi Halaman Kelola Data Produk **Error! Bookmark not defined.**

Gambar 6.4 Hasil Implemenasi Halaman Kelola Data Penjualan **Error! Bookmark not defined.**

Gambar 6.5 Hasil Implementasi Halaman Beranda Pelanggan **Error! Bookmark not defined.**

Gambar 6.6 Hasil Implementasi Halaman Rincian Belanja **Error! Bookmark not defined.**

Gambar 7.1 Simbol Compatibility Testing pada SortSite **Error! Bookmark not defined.**

Gambar 7.2 Hasil *Compatibility Testing* pada SortSite **Error! Bookmark not defined.**

DAFTAR LAMPIRAN

LAMPIRAN A HASIL WAWANCARA.....Error! Bookmark not defined.

LAMPIRAN B HASIL PERULANGAN PROTOTYPING....Error! Bookmark not defined.

LAMPIRAN C HASIL DOKUMEN UAT.....Error! Bookmark not defined.

C.1 UAT untuk Manager.....Error! Bookmark not defined.

C.2 UAT untuk PenjualError! Bookmark not defined.

C.3 UAT untuk KasirError! Bookmark not defined.





BAB 1 PENDAHULUAN

1.1 Latar belakang

Perkembangan teknologi yang sangat pesat menciptakan banyak perubahan pada saat ini. Dampak yang sangat besar dapat dilihat dari perusahaan yang semakin berkembang tak terkecuali perusahaan kecil sekalipun. Informasi menjadi bagian penting dalam perusahaan. Aliran informasi yang cepat sangat membantu kinerja perusahaan, sehingga aliran informasi dalam perusahaan dapat tersampaikan dengan benar dan baik. Perusahaan memiliki informasi-informasi yang perlu disampaikan dengan cepat. Salah satu media yang digunakan dalam membantu perpindahan aliran informasi dalam perusahaan adalah internet. Teknologi internet dapat membantu aliran informasi di perusahaan tersampaikan lebih cepat, lebih baik dan juga lebih merata.

Internet adalah sebuah jaringan yang dapat diakses secara global di seluruh dunia. Melalui jaringan komputer yang saling berhubungan, Internet memungkinkan individu dan bisnis untuk berbagi informasi, sumber daya, dan layanan (Reid, 2008). Penggunaan internet pada perusahaan-perusahaan sekarang dibutuhkan untuk sarana komunikasi, baik dengan pelanggan maupun dengan pegawai perusahaan itu sendiri. Salah satu perusahaan yang membutuhkan komunikasi antara pegawai dan pelanggan adalah perusahaan dalam bidang penjualan. Perusahaan dalam bidang penjualan memerlukan informasi secara real time. Perusahaan ini membutuhkan internet untuk berkomunikasi dengan pelanggan agar memudahkan pelanggan mendapatkan informasi dan tetap menjalin hubungan dengan perusahaan. Kemudahan pelanggan untuk mengetahui tentang barang yang dimiliki perusahaan juga diperlukan oleh pelanggan.

Branded IT Store merupakan perusahaan yang memiliki 2 cabang toko yang terletak di Malang dan bergerak di bidang penjualan elektronik ini belum memiliki sistem yang mampu memberikan kemudahan bagi pelanggan untuk mengetahui produk yang dimiliki perusahaan secara online serta sistem yang dapat melakukan pencatatan secara otomatis terkait dengan data pelanggan. Pencatatan masih dilakukan secara manual dan tidak ada tindak lanjut terkait dengan catatan yang ada. Penjualan yang terjadi setiap hari membutuhkan pengelolaan data secara *real time*. Branded IT Store melayani pembelian hanya melalui toko dalam skala kecil hingga pembelian dalam skala besar berupa, laptop, printer serta komputer. Branded IT Store memiliki beberapa pegawai yang memiliki *shift* kerja yang berbeda beda. Penyampaian informasi antara pegawai masih belum terintegrasi dengan baik, yaitu dengan melakukan pencatatan secara manual yang membuat penyampaian informasi tidak tersampaikan secara langsung. Ketika pembeli melakukan pembelian dengan membayar DP, transaksi tersebut dicatat oleh pegawai secara manual. Pembeli hanya mendapatkan sebuah catatan yang berupa informasi tentang jumlah pembayaran DP dan tanggal. Informasi terkait produk dan harga yang akan dibayarkan tidak tercantum.

Hal tersebut mengakibatkan barang yang sudah dipesan oleh pembeli dapat sewaktu-waktu terjual kepada pembeli yang berbeda, karena pegawai penjualan lain tidak mengetahui mengenai pemesanan yang sudah terjadi. Pegawai penjualan yang tidak mengetahui hal tersebut akan langsung melakukan proses transaksi terhadap pembeli yang baru. Hal ini akan menimbulkan kekecewaan bagi pemesan yang telah melakukan pemesanan dan pembayaran DP terlebih dahulu. Permasalahan juga terjadi ketika ada perubahan harga terhadap produk yang sudah dibayar DP. Ketika produk yang sudah dibayar DP akan dibayar lunas dan terjadi perubahan harga dengan yang sudah disepakati diawal, tidak ada bukti transaksi yang menunjukkan harga barang yang sudah disepakati sebelumnya. Pada bukti transaksi pembayaran DP tidak tercantum harga produk yang akan dibeli dan tenggat waktu pembayaran yang diberikan pada produk yang sudah dibayar DP.

Salah satu modul operasional *Customer Relationship Management* (CRM) yang diterapkan adalah *Sales Force Automation* (SFA). *Sales Force Automation* merupakan sebuah sistem yang dapat mengintegrasikan penjualan dan pelayanan pada perusahaan yang bertujuan untuk meningkatkan kepuasan pelanggan, terutama untuk memperoleh pelanggan baru dan mempertahankan pelanggan yang telah ada (Buttle, 2009). *Salesforce* adalah sistem berbasis internet yang tidak membutuhkan instalasi *software* sehingga hanya cukup *login* pada situs *Web*, maka perusahaan sudah dapat mulai mengelola setiap kegiatan yang ada. Dengan *Salesforce*, maka perusahaan dapat mengelola mitra usaha dan pelanggan. (Wong, 2007). Dengan sistem ini diharapkan dapat membantu perusahaan dalam memberikan informasi terkait produk yang dimiliki dan melakukan proses penjualan secara lebih luas serta dapat meningkatkan kualitas pelayanan terhadap pelanggan guna memberi kenyamanan dan mampu meningkatkan proses bisnis. Pengembangan sistem informasi *Sales Force Automation* tersebut dilakukan dengan menggunakan metode *prototyping* untuk memberikan pemahaman mengenai *Sales Force Automation* kepada pelanggan. Metode *prototyping* dipilih karena dapat memudahkan pelanggan dalam memahami persyaratan kebutuhan yang dibutuhkan pelanggan (Verma, 2014).

Oleh karena itu, penulis melakukan penelitian yang berjudul “Pengembangan Sistem Informasi *Sales Force Automation* (SFA) dengan menggunakan metode *Prototyping* pada Branded IT Store Malang”. Diharapkan dengan adanya penelitian ini dapat memberikan solusi untuk masalah-masalah diatas.

1.2 Rumusan masalah

Berdasarkan latar belakang di atas, maka rumusan masalah pada skripsi ini adalah sebagai berikut:

1. Apa saja kebutuhan fungsional dan non-fungsional pada sistem informasi sistem informasi dengan modul *Sales Force Automation* (SFA) untuk proses penjualan pada kasus Branded IT Store Malang?

2. Bagaimana hasil pengembangan *prototyping* pada sistem informasi dengan modul *Sales Force Automation* (SFA) untuk proses penjualan pada kasus Branded IT Store Malang?
3. Bagaimana hasil pengujian validasi, *compatibility*, dan *user acceptance testing* pada sistem informasi dengan modul *Sales Force Automation* (SFA) untuk proses penjualan pada kasus Branded IT Store Malang?

1.3 Tujuan

Adapun tujuan dari penelitian ini yaitu :

1. Menentukan kebutuhan fungsional dan non-fungsional pada sistem informasi *Sales Force Automation* (SFA) untuk proses penjualan pada kasus Branded IT Store Malang.
2. Mengembangkan sistem informasi dengan modul *Sales Force Automation* (SFA) menggunakan metode *prototyping* untuk proses penjualan pada kasus Branded IT Store Malang.
3. Melakukan pengujian terhadap sistem informasi dengan modul *Sales Force Automation* (SFA) untuk proses penjualan pada kasus Branded IT Store Malang dengan pengujian validasi, *compatibility testing*, dan *user acceptance testing*.

1.4 Manfaat

Manfaat yang diperoleh dengan penelitian ini antara lain :

1. Bagi perusahaan, diharapkan Branded IT Store nanti akan memiliki sistem informasi dengan modul *Sales Force Automation* (SFA) untuk proses penjualan yang dapat menangani pengelolaan data pelanggan menangani penjualan sehingga tidak terjadi salah paham antara penjual dan pelanggan, serta memiliki sebuah katalog online yang dapat memudahkan pelanggan dalam bertransaksi sehingga dapat memperluas jaringan hubungan dengan pelanggan.
2. Bagi penulis dan pihak terkait dengan pembuatan sistem informasi, hasil penelitian dapat dijadikan bahan wacana dan rujukan penelitian lebih lanjut.

1.5 Batasan masalah

Berdasarkan pada latar belakang dan rumusan masalah yang telah dipaparkan di atas, terdapat batasan masalah pada penelitian ini, diantaranya:

1. Sistem yang dibuat berbasis web untuk memudahkan akses pengguna.
2. Sistem yang dirancang menerapkan fungsi yang ada pada *Sales Force Automation* (SFA).
3. Sistem yang dirancang meliputi pengelolaan data produk, pengelolaan data penjualan, pengelolaan data pelanggan dan pengelolaan data pegawai.

4. Prototyping akan dikembangkan hingga mencapai hasil yang paling mendekati dengan spesifikasi kebutuhan fungsional pengguna.

1.6 Sistematika pembahasan

BAB I PENDAHULUAN

Pada bab pendahuluan menjelaskan latar belakang penulisan, permasalahan yang dihadapi, batasan masalah, tujuan, manfaat, serta sistematika penulisan skripsi.

BAB II LANDASAN KEPUSTAKAAN

Pada bab ini menjelaskan tentang dasar-dasar teori yang mendasari masalah yang diteliti.

BAB III METODOLOGI PENELITIAN

Pada bab ini menjelaskan metode atau langkah-langkah yang digunakan dalam penulisan skripsi dan perancangan sistem yang akan dibangun.

BAB IV ANALISIS KEBUTUHAN

Pada bab ini menjelaskan gambaran dari kebutuhan sistem dengan melakukan pemodelan kondisi proses bisnis Branded IT Store pada saat ini dan proses bisnis usulan serta memberikan deskripsi umum perangkat lunak, identifikasi, spesifikasi kebutuhan dan pemodelan *use case* yang akan menjadi landasan dalam perancangan sistem.

BAB V PERANCANGAN

Pada bab ini menjelaskan bagaimana perancangan sistem yang akan dibangun.

BAB VI IMPLEMENTASI

Pada bab ini melakukan pengimplementasian perangkat lunak menyesuaikan dengan perancangan yang telah dibuat.

BAB VII PENGUJIAN & EVALUASI

Pada bab ini berisikan hasil dari pengujian sistem informasi *Sales Force Automation* dengan tujuan untuk mengetahui hasil sistem yang dikembangkan sudah sesuai dengan yang dibutuhkan.

BAB VIII KESIMPULAN DAN SARAN

Pada bab ini berisikan kesimpulan dan saran yang diperoleh melalui penelitian serta pemecahan masalah yang dapat dilakukan pada pengembangan selanjutnya.



BAB 2 LANDASAN KEPUSTAKAAN

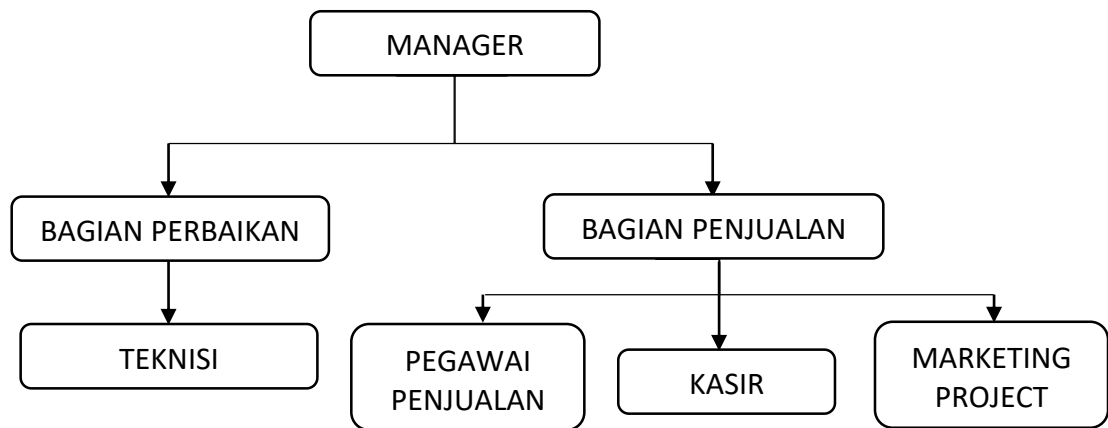
2.1 Kajian Pustaka

Penelitian yang berkaitan dengan pengembangan sistem informasi *sales force automation* dengan judul “Analisa dan Perancangan *Customer Relationship Management* (CRM) Sub Domain *Sales Force Automation* dan *Reporting Automation* dalam Bisnis *Clothing Line*”. Dalam penelitiannya pada The Pantjing Apparel, The Pantjing Apparel sama sekali belum menggunakan pencatatan secara otomatis serta histori pemesanan yang telah dilakukan masih belum masuk kedalam pencatatan mereka. Pencatatan masih dilakukan secara manual dan tidak ada tindak lanjut apapun akan catatan manual tersebut. Dengan demikian dibuatlah analisis dan perancangan *customer relationship management* yang dapat membantu menyelesaikan permasalahan tersebut. Perbedaan penelitian sebelumnya berada pada penyelesaian masalah yang hanya sebatas analisa dan perancangan. Penelitian sebelumnya tidak mengembangkan sistem informasi sampai dapat digunakan.

Penelitian lain yang berkaitan dengan judul “Aplikasi *Sales Force Automation* dengan Studi Kasus Pada CV. Blessia Garmino”. Pada penelitian tersebut, menjelaskan bagaimana penerapan *sales force automation* dalam pengembangan sebuah sistem. Pengembangan sistem *sales force automation* berfokus pada otomatisasi kegiatan penjualan. Dari hasil analisis penelitian yang didapat, aplikasi yang dibuat dapat membantu kegiatan penjualan dari CV. Blessia Garmino. Kegiatan penjualan dapat berupa pengelolaan data penjualan dan pengelolaan pegawai.

2.2 Profil Perusahaan

Branded IT Store merupakan perusahaan yang terletak di jalan Bendungan Sutami 58B, Malang. Anak perusahaan dari PT. MEGASTAR aktif bergerak dalam penjualan barang elektronik seperti komputer, laptop, printer, aksesoris komputer, perbaikan berbagai macam laptop dan sebagai *authorized HP service*. Branded IT Store sebelumnya terletak di Mall Dieng yang sekarang bernama Cyber Mall. Adanya perubahan kebijakan yang terjadi di Mall Dieng menyebabkan Branded IT Store membeli salah satu ruko yang berada di Bendungan Sutami Malang. Branded IT Store sendiri terbentuk karenaantisipasi dan memperluas jaringan dari PT. MEGASTAR di daerah Malang. Pelanggan yang dimiliki oleh Branded IT Store berasal dari berbagai kalangan, mulai dari anak sekolah, mahasiswa, perkantoran, bahkan perusahaan sebesar Telkom merupakan pelanggan dari Branded IT Store Malang.



Gambar 2.1 Struktur Organisasi Branded IT Store Malang

*Keterangan:

Manager: Meri Oen

Teknisi: Christian

Pegawai Penjualan:

1. Nisa
2. Rahman
3. Samsul
4. Tefan

Kasir:

1. Yudia
2. Dewi

Marketing Project:

1. Anas
2. Anas

2.3 Customer Relationship Management

2.3.1 Pengertian Customer Relationship Management

Customer Relationship Management memiliki arti yang luas. Perbedaan pendapat dari tiap pakar atau berbagai sumber memberikan banyak alternatif pengertian CRM. Definisi yang diberikan tergantung pada sudut pandang yang mereka lihat.

Salah satu dari banyak alternatif pengertian CRM, deskripsi tentang CRM sebagai berikut:

CRM adalah inti dari sebuah bisnis strategi yang bertujuan untuk membuat dan memberikan nilai lebih kepada pelanggan tertentu untuk mendapat keuntungan. CRM mengintegrasikan proses internal dan fungsi-fungsi antara departemen

dalam perusahaan yang dapat memberikan kemudahan pada perusahaan menjalankan bisnisnya. Akses ke data pelanggan yang bersangkutan, memungkinkan bagian penjualan, bagian pemasaran dan bagian layanan mengetahui interaksi satu sama lain dengan pelanggan(Buttle, 2009).

CRM merupakan sebuah pendekatan strategis yang berkaitan dengan menciptakan peningkatan nilai pemegang saham melalui pengembangan hubungan dengan pelanggan utama dan segmen pelanggan. CRM menyatukan potensi penggunaan IT dan strategi hubungan pemasaran untuk memberikan keuntungan dan hubungan jangka panjang. Khususnya, CRM menyediakan kesempatan untuk menggunakan data dan informasi untuk memahami pelanggan dan menerapkan strategi hubungan pemasaran yang lebih baik. Hal ini membutuhkan integrasi *cross-functional*, operasi, proses dan kemampuan pemasaran yang bisa didapatkan melalui informasi, teknologi dan aplikasi(Payne, 2005).

3 komponen penting CRM(Gray, 2001) dijelaskan sebagai berikut:

a. Customer

Customer atau pelanggan adalah satu-satunya sumber keuntungan dan pengembangan selanjutnya bagi perusahaan. Pelanggan yang baik, yang memberikan keuntungan lebih sulit untuk didapatkan, karena pelanggan ini memahami produk perusahaan dan para kompetitor lainnya. Teknologi informasi dapat membantu untuk memisahkan dan mengelola pelanggan dengan baik. CRM dapat dijadikan sebagai pendekatan pemasaran berdasarkan informasi pelanggan.

b. Relationship

Hubungan antara perusahaan dan pelanggan memiliki peran dalam komunikasi dan interaksi antar keduanya. Hubungan ini bisa menjadi hubungan jangka panjang maupun jangka pendek, lanjut atau sementara, dan berulang atau sekali waktu. Walaupun pelanggan memiliki sikap yang baik terhadap perusahaan dan produknya, kebiasaan membeli mereka tergantung pada situasi.

c. Management

CRM bukan hanya aktivitas dalam pemasaran. Informasi pelanggan yang dikumpulkan dijadikan keuntungan bagi pasar.

2.3.2 Tujuan dan Manfaat CRM

CRM sendiri memiliki beberapa tujuan, tujuan-tujuannya sebagai berikut(Kalakota & Robinson, 2001):

1. Menggunakan hubungan yang sudah ada untuk meningkatkan pendapatan. Hal ini berarti mempersiapkan pandangan yang komprehensif dari pelanggan untuk memaksimalkan hubungan mereka dengan

perusahaan baik melalui *up-selling* atau *cross-selling* dan pada saat yang sama, meningkatkan profit, menarik perhatian pelanggan dan mempertahankan pelanggan baik.

2. Menggunakan informasi yang terintegrasi untuk pelayanan yang memuaskan. Dengan menggunakan informasi dari pelanggan untuk meningkatkan pelayanan yang lebih baik berdasarkan kebutuhan mereka, perusahaan dapat menghemat waktu pelanggan dan menyingkirkan segala kekecewaan pelanggan. Sebagai contoh, pelanggan seharusnya tidak mengulangi informasi yang sama ke berbagai departemen perusahaan secara terus-menerus. Pelanggan akan senang jika pihak perusahaan tahu banyak tentang apa yang mereka inginkan.
3. Menciptakan saluran proses dan prosedur komunikasi yang konsisten dan berulang.

Melalui tujuan-tujuan CRM yang disebutkan diatas dapat disimpulkan manfaat apa saja yang didapat dari tujuan diatas, antara lain:

1. Jumlah konsumen bertambah, yaitu mencari konsumen baru di samping tetap memelihara tingkat kepuasan konsumen yang sudah ada.
2. Mengetahui tingkat kepemilikan perusahaan pada konsumen, yaitu dengan mengetahui kebutuhan konsumen.
3. Mengetahui kebutuhan konsumen pada masa yang akan datang, yaitu melalui hasil transaksi yang sudah dilakukan dan dari hasil analisis data transaksi yang sudah terkumpul.
4. Mengetahui perbaikan yang harus dilakukan pada layanan yang diberikan kepada konsumen.
5. Mampu menganalisis pola data transaksi, sebagai contoh mampu mengetahui kombinasi produk yang akan dijual pada waktu-waktu tertentu.
6. Mengurangi resiko operasional, yaitu dengan mengetahui prediksi yang akan terjadi dan kesalahan yang pernah dilakukan melalui customer history.

2.3.3 Jenis-Jenis CRM

CRM dapat diklasifikasikan kedalam beberapa tipe, diantaranya:

1. Operasional CRM

Operasional CRM berhubungan dengan otomatisasi proses bisnis yang berkaitan dibagian *front-office*. Mengotomatisasi dan meningkatkan kemampuan menghadapi pelanggan. Operasional CRM memungkinkan bagian pemasaran, penjualan dan layanan berjalan secara otomatis dan terintegrasi. Ada beberapa jenis otomatisasi diantaranya:

- a. *Marketing Automation (MA)*

MA adalah proses otomatis yang diaplikasikan dalam proses pemasaran. Proses ini memungkinkan penjual untuk menggunakan data pelanggan yang terkait dalam rangka mengembangkan, melaksanakan dan mengevaluasi komunikasi dan penawaran yang ditargetkan (Buttle, 2009). MA memudahkan perusahaan untuk melakukan proses pemasaran dengan memberikan informasi kepada pelanggan yang bersifat promosi.

b. Sales-Force Automation (SFA)

Secara umum SFA dapat diartikan sebagai penerapan teknologi komputerisasi untuk mendukung orang-orang pada bagian penjualan dan manajemen penjualan dalam pencapaian tujuan yang berhubungan dengan pekerjaan mereka. Sebuah perangkat lunak SFA dirancang sehingga data pelanggan yang berhubungan dapat diambil, disimpan, dianalisis dan dibagikan kepada orang-orang bagian penjualan dan manajer agar menjadi lebih efektif atau efisien dalam pencapaian tujuan mereka (Buttle, 2009). SFA memudahkan pengawasan alur penjualan dan membantu mengoptimalkan penyampaian informasi antara penjual.

c. Service Automation (SA)

SA berbeda secara signifikan, tergantung pada produk yang dilayani. SA memungkinkan perusahaan untuk mengelola operasi layanan mereka, melalui *call center*, *web* atau bertatap muka secara langsung. SA memudahkan perusahaan untuk menangani dan mengkoordinasikan layanan terkait dengan komunikasi *inbound* dan *outbound* melalui semua saluran. Hal ini memungkinkan pengguna untuk menjadi lebih efisien dan efektif dengan mengurangi biaya jasa, menambah kualitas layanan, mengangkat produktivitas dan meningkatkan kepuasan pelanggan (Buttle, 2009). SA akan mengelola masukan dari pelanggan, berupa kritik, saran maupun keluhan atas pelayanan dari suatu perusahaan.

2. *Strategic CRM*

Budaya bisnis *customer-centric*. Budaya ini didedikasikan untuk mendapatkan dan mempertahankan pelanggan dengan memberikan nilai yang lebih baik dibanding dengan kompetitor lain. Budaya ini ditunjukkan dalam perilaku kepemimpinan dan desain sistem formal perusahaan (Buttle, 2009).

3. *Analytical CRM*

Berhubungan dengan penyimpanan, analisis, interpretasi dan penggunaan data yang dibuat dari sisi operasionalnya. *Analytical CRM* berfokus pada proses penggunaan logika dan rumus perhitungan untuk melakukan pengolahan data pelanggan yang dimiliki oleh perusahaan (Payne, 2005).

2.3.4 Pengertian *Sales Force Automation (SFA)*

SFA mengacu pada sistem penjualan dan pemasaran yang dimuat didalam komputer untuk menghubungkan tenaga penjualan. Sehingga memungkinkan penjualan produk dapat langsung dilaporkan dan dipesan melalui sistem. Keuntungan dari penggunaan SFA adalah dapat mengetahui aktivitas layanan pelanggan dan penjualan terhadap pelanggan secara bersama (Payne, 2005).

SFA juga dapat digunakan untuk mengawasi alur penjualan yang ada dan mengoptimalkan penyampaian informasi dengan adanya integrasi informasi antara bagian penjualan. SFA adalah sistem yang secara otomatis mencatat semua tahap dalam proses penjualan. Ada beberapa fungsi yang ditawarkan oleh SFA diantaranya, manajemen pemesanan (*order management*), manajemen kontak (*contact management*), manajemen data pelanggan (*account management*), manajemen dan aktivitas (*activity management*) (Buttle, 2009).

1. Manajemen Pemesanan (*Order Management*)

Manajemen pemesanan digunakan untuk memudahkan proses pemesanan produk oleh pelanggan sehingga memungkinkan pesanan didapat lebih cepat.

2. Manajemen Kontak (*Contact Management*)

Manajemen kontak digunakan untuk mengelola seluruh data milik pelanggan yang masuk ke dalam database dan termasuk pemesanan dari tiap pelanggan yang pernah terjadi.

3. Manajemen Akun (*Account Management*)

Manajemen akun digunakan untuk memudahkan pegawai mengelola data pelanggan yang dimiliki perusahaan untuk menawarkan produk kembali.

4. *Product Encyclopedias*

Product encyclopedias menyediakan sebuah katalog yang dapat memudahkan pegawai penjualan untuk memberikan penawaran produk yang tersedia. Fungsi ini juga dapat diakses oleh pelanggan untuk mengetahui produk dari perusahaan.

Sales force automation merupakan sebuah sistem yang dapat membantu untuk otomatisasi kegiatan yang biasa dikerjakan oleh seseorang yang bertugas dalam kegiatan penjualan hingga tercapai tujuan dari proses penjualan tersebut. Kemudahan yang diberikan melalui SFA dapat berupa pengelolaan data pelanggan yang memudahkan pegawai untuk menganalisis karakter pelanggan dan pengelolaan penjualan dimana proses penjualan dapat menjadi lebih efektif. SFA juga memiliki fungsi *product encyclopedias* yang dapat memudahkan pelanggan dan pegawai dalam penyampaian produk yang dimiliki perusahaan.

2.4 Proses Bisnis

Proses bisnis adalah sebuah kumpulan aktivitas yang dirancang untuk menghasilkan *output* untuk pelanggan atau pasar tertentu (Sparx System, 2004). Proses bisnis memiliki komponen yaitu : (1) memiliki tujuan yang jelas, (2) adanya input atau masukan, (3) adanya output atau keluaran, (4) menggunakan resource. (5) mempunyai sejumlah kegiatan yang terbagi dalam beberapa tahapan, (6) dapat memberikan pengaruh terhadap lebih dari satu unit dalam organisasi dan (7) dapat menciptakan nilai atau *value* bagi konsumen.

2.5 Business Process Model Notation

Business Process Modelling Notation (BPMN) adalah representasi berupa grafis yang berguna untuk menentukan proses bisnis dan aktor yang terlibat dalam proses tersebut. *Business Process Modelling Notation* (BPMN) mempunyai tujuan untuk menyediakan notasi yang dapat dipahami oleh pengguna bisnis, berawal dari analisis bisnis yang membuat rancangan proses, lalu orang yang menjalankan proses atau yang mengimplementasikan teknologi, serta orang bisnis yang mengelola dan mengontrol proses tersebut (Silver, 2012).

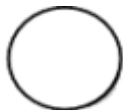


Ada 4 kategori elemen yang dimiliki *Business Process Modelling Notation* (BPMN) yang berguna sebagai penunjang dalam pembuatan *Business Process Modelling Notation* (BPMN). Empat kategori elemen tersebut adalah (Object Management Group, 2011):

2.5.1 Flow Object

1. Event

Event merupakan sebuah proses yang terjadi dan memiliki pengaruh dalam proses bisnis. Suatu *event* dapat berasal dari dalam dan luar suatu proses. Terdapat 3 tipe *Event* yaitu *start event*, *intermediate event*, dan *end event*. Tabel 2.1 merupakan penjelasan dari tipe *event*.



Tabel 2.1 Simbol-simbol pada tipe event

No	Tipe	Deskripsi	Simbol
1	<i>Start</i>	Mendeskripsikan dimana suatu proses dimulai.	
2	<i>Intermediate</i>	Mendeskripsikan dimana sesuatu terjadi diantara awal dan akhir proses. Akan mempengaruhi alur dari proses, tapi tidak akan memulai atau memberhentikan proses	
3	<i>End</i>	Mendeskripsikan suatu proses berakhir	

2. Activity

Activity merupakan notasi yang mempunyai kegunaan untuk merepresentasikan sebuah pekerjaan yang di ada dalam sebuah proses. 2 tipe yang dimiliki oleh *Activity* yaitu *sub-process* dan *task* pada tabel 2.2.



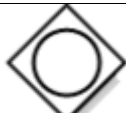

Tabel 2.2 Simbol-simbol pada tipe *activity*

No	Tipe	Deskripsi	Simbol
1	<i>Task</i>	Merupakan aktivitas yang dilakukan pada alur proses.	
2	<i>Sub Process</i>	Merupakan sebuah aktivitas yang bersifat majemuk yang dimasukkan ke dalam proses. Aktivitas majemuk tersebut dapat dijabarkan dengan lebih <i>detail</i> .	

3. Gateway

Gateway bertanggung jawab mengontrol bagaimana alur dari sebuah proses bisnis. Tabel 2.3 merupakan penjelasan dari tipe *gateway*




Tabel 2.3 Simbol-simbol pada tipe *gateway*

No	Tipe	Deskripsi	Simbol
1	<i>Exclusive</i>	Memiliki kegunaan untuk membuat alur alternatif pada proses bisnis. Melalui beberapa alur alternatif yang dibuat, hanya 1 alur alternatif yang dapat dijalankan sesuai dengan kondisi yang terpenuhi	
2	<i>Parallel</i>	Mendeskripsikan proses yang dijalankan secara bersamaan.	
3	<i>Inclusive</i>	Mendeskripsikan sebuah proses yang di pecah menjadi beberapa jalur.	
4	<i>Complex</i>	Mendeskripsikan alur yang kompleks pada sebuah proses bisnis.	

2.5.2 Connections

Connections merupakan notasi yang menghubungkan *flow objects*. Tabel 2.4 merupakan penjelasan dari tipe *connections*.



Tabel 2.4 Simbol-simbol pada tipe *connections* (lanjutan)

No	Tipe	Deskripsi	Simbol
1	<i>Sequence Flow</i>	Menunjukkan urutan aktivitas yang dijalankan pada sebuah proses.	
2	<i>Association</i>	Menunjukkan hubungan antara data, teks, artifak lain, dan <i>flow object</i> pada sebuah proses.	
3	<i>Message Flow</i>	Menunjukkan alur pesan antara dua partisipan yang mampu mengirim dan menerima pesan.	

2.5.3 Swimlanes

Swimlanes merupakan wadah grafis yang membagi suatu set aktivitas dengan aktivitas lain. Tabel 2.5 merupakan penjelasan tipe swimlanes.



Tabel 2.5 Simbol-simbol pada tipe *swimlanes*

No	Tipe	Deskripsi	Simbol
1	<i>Pool</i>	Merupakan wadah yang berisi satu proses dan <i>sequence flow</i> yang menghubungkan aktivitas.	
2	<i>Lane</i>	Digunakan untuk merepresentasikan tanggung jawab aktivitas pada sebuah proses.	



2.5.4 Artifacts

Artifacts merupakan notasi sebuah objek diluar sebuah proses. Artifacts dapat merepresentasikan data atau catatan yang menjelaskan sebuah proses atau dapat digunakan untuk mengelola tugas atau proses. Tabel 2.6 merupakan penjelasan tipe artifacts.

Tabel 2.6 Simbol-simbol pada tipe *artifacts*

No	Tipe	Deskripsi	Simbol
1	<i>Data Object</i>	Memiliki kegunaan untuk memberikan informasi yang mengalir pada sebuah proses seperti dokumen bisnis, surat, <i>email</i> , dan lain-lain.	
2	<i>Data Store</i>	Tempat dimana proses dapat membaca atau menulis data.	

Tabel 2.6 Simbol-simbol pada tipe *artifacts* (lanjutan)

No	Type	Deskripsi	Simbol
3	<i>Annotation</i>	Menunjukkan informasi tambahan kepada pembaca sebuah diagram <i>Business Process Model Notation</i> (BPMN).	
4	<i>Group</i>	Memungkinkan untuk mengelompokkan elemen secara informal.	

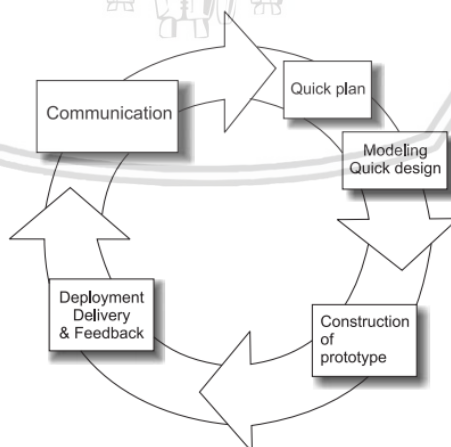
2.6 Software Development Life Cycle (SDLC)

SDLC adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan untuk mengembangkan sistem perangkat lunak (Shalahudin, 2015).

Pada *framework* SDLC terdapat beberapa metode, salah satunya merupakan metode *prototyping* yang akan peneliti pakai untuk pengembangan sistem informasi *sales force automation*, pada sub-bab berikut ini akan dijelaskan mengenai metode *prototyping*.

2.6.1 Model *Prototyping*

Model *prototyping* dimulai dari mengumpulkan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat. Salah satu kesulitan utama pengguna adalah untuk mengungkapkan kebutuhan mereka secara tepat. Model *prototyping* dapat digunakan untuk menyambungkan ketidakpahaman pelanggan mengenai hal teknis dan memperjelas spesifikasi kebutuhan yang diinginkan pelanggan (Vilet, 2007).



Gambar 2.2 Prototyping sebagai alat untuk kebutuhan perangkat lunak

Sumber: Agarwal (2010)

Model *prototyping* merupakan pilihan tepat untuk menggambarkan kebutuhan pelanggan secara lebih detail karena pelanggan seringkali mengalami kesulitan dalam menyampaikan kebutuhannya secara detail tanpa melihat

gambaran yang jelas. Permasalahan sering terjadi pada model *prototyping*, salah satunya adalah iteratif yang terlalu banyak. Permasalahan tersebut dapat diatasi dengan melakukan perjanjian antara pengembang dan pelanggan agar model *prototyping* hanya digunakan untuk mendefinisikan spesifikasi kebutuhan perangkat lunak, tapi tidak untuk keseluruhan proses pengembangan sistem perangkat lunak. Agar proyek dapat berjalan sesuai dengan target, untuk mengantisipasi sebaiknya spesifikasi kebutuhan perangkat lunak harus sudah disepakati oleh pengembang dengan pelanggan secara tertulis. Dokumen tersebut akan menjadi batasan agar spesifikasi kebutuhan sistem masih dalam ruang lingkup pengembangan (Shalahudin, 2015).

Tujuan penting penggunaan *prototyping* adalah untuk menggambarkan format data masukan, pesan, laporan, dan dialog interaktif kepada pelanggan. Hal tersebut sangat baik untuk mendapatkan pemahaman yang lebih tentang kebutuhan pelanggan (Agarwal, 2010). Model prototipe sangat berguna dalam mengembangkan *Graphical User Interface* (GUI) bagian dari sebuah sistem.

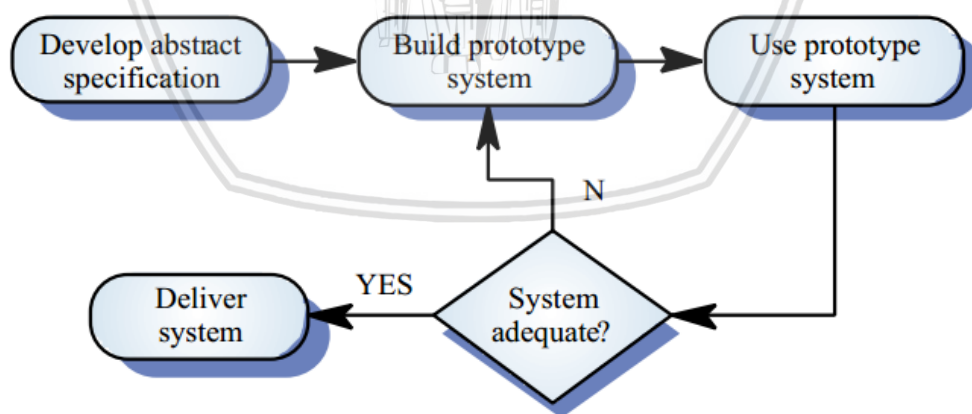
Prototyping terbagi menjadi 2 proses diantaranya (Sommerville, 2000):

1. *Evolutionary Prototyping*

Pendekatan untuk pengembangan sistem dimana *prototyping* awal diproduksi dan disempurnakan melalui sejumlah tahap sampai sistem akhir.

2. *Throw-Away Prototyping*

Sebuah *prototyping* yang biasanya merupakan implementasi praktis dari sistem yang diproduksi untuk membantu menemukan masalah persyaratan dan kemudian dibuang. Sistem ini kemudian dikembangkan dengan menggunakan beberapa proses pengembangan lainnya.



Gambar 2.3 Diagram Pengembangan Evolutionary Prototyping

Sumber: Sommerville (2000)

Peneliti akan menggunakan *prototyping* dengan pendekatan *Evolutionary Prototyping*. *Evolutionary Prototyping* adalah pendekatan pengembangan di mana bagian-bagian tertentu dari sistem terlebih dahulu dikembangkan dan kemudian mengembangkan keseluruhan sistem dari bagian-bagian tersebut. Berbeda dengan jenis *prototyping* yang lain, dalam *Evolutionary Prototyping* kode

prototyping tidak dibuang melainkan dikembangkan kembali menjadi kode yang akan diberikan diakhir(Sommerville, 2000).

Pada gambar 2.3 menjelaskan bagaimana *evolutionary prototyping* dikembangkan. Mulai dari mengembangkan spesifikasi di mana pengembang menganalisis spesifikasi kebutuhan pelanggan untuk selanjutnya membangun sebuah sistem *prototyping*. *Prototyping* yang sudah dibangun akan disampaikan ke pelanggan. Jika *prototyping* yang dibangun sudah memadai dengan spesifikasi yang didapatkan melalui analisis diawal pengembangan maka sistem akan dilakukan pengujian sebelum diserahkan kepada pelanggan. Jika *prototyping* masih belum memadai, maka pengembang akan membangun kembali *prototyping* dengan memperbaiki sistem *prototyping* yang sudah dibangun sebelumnya.

2.7 Unified Modeling Language (UML)

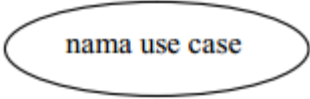
Pada umumnya, UML digunakan untuk pemodelan perangkat lunak. UML adalah bahasa standar untuk menulis *blueprints* perangkat lunak. UML dapat digunakan untuk memvisualisasikan, menentukan, membuat, dan mendokumentasikan arsitektur sistem perangkat lunak(Rumbaugh, Jacobson, & Booch, 1998). UML memvisualisasikan dengan diagram untuk memberikan pandangan yang jelas tentang sistem yang akan dibuat. Beberapa diagram untuk memvisualisasikan adalah sebagai berikut:

- *Use Case Diagram*
- *Activity Diagram*
- *Sequence Diagram*
- *Class Diagram*

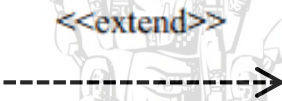
2.7.1 Use Case Diagram

Use-case diagram sangat penting untuk memodelkan perilaku suatu sistem, subsistem, atau kelas. Masing-masing *use case* menunjukkan deskripsi fungsi dengan aktor pengguna dan hubungan mereka(Rumbaugh, Jacobson, & Booch, 1998). *Use-case* menggambarkan sebuah fungsionalitas dari suatu sistem, sehingga pengguna sistem harus memahami dan mengerti mengenai sistem yang akan dibuat. Adapun simbol-simbol yang terdapat dalam *use-case* diagram terdapat pada tabel 2.7

Tabel 2.7 Simbol-Simbol Use Case Diagram

No	Nama	Gambar	Fungsi
1	<i>Uce Case</i>		Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan

Tabel 2.7 Simbol-Simbol Use Case Diagram (lanjutan)






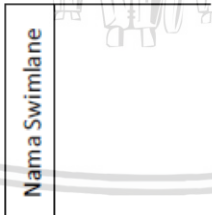
No	Nama	Gambar	Fungsi
			menggunakan kata kerja di awal frase nama <i>use case</i>
2	<i>Actor</i>		Orang, proses, atau sistem lain yang dapat berinteraksi dengan sistem yang akan dibuat. Walaupun simbol dari aktor adalah gambar orang akan tetapi belum tentu merupakan orang. Biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
3	<i>Association</i>		Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4	<i>Extend</i>		Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.
5	<i>Generalization</i>		Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya
6	<i>Include</i>		Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini

Sumber: A.S & Shalahuddin [2011]

2.7.2 Activity Diagram

Activity diagram tidak hanya penting untuk memodelkan aspek dinamik suatu sistem, tetapi juga untuk membangun sistem yang dapat dieksekusi melalui teknik maju dan mundur. Diagram aktivitas pada dasarnya adalah *flowchart*, yang menunjukkan aliran kontrol dari aktivitas ke aktivitas (Rumbaugh, Jacobson, & Booch, 1998). Simbol-simbol dari *activity diagram* terdapat pada tabel 2.8.

Tabel 2.8 Simbol-Simbol Activity Diagram

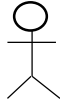




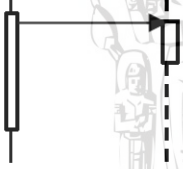
No	Nama	Gambar	Fungsi
1	Status Awal		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki status awal
2	Aktivitas		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3	Decision		Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
4	Join (Penggabungan)		Asosiasi <i>join</i> dimana lebih dari satu aktivitas digabungkan menjadi satu
5	Status Akhir		Status Akhir yang dilakukan sistem, sebuah diagram memiliki sebuah status akhir
6	Swimlane		Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Sumber: A.S & Shalahuddin [2011]

2.7.3 Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri antar dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait) (Dharwiyanti & Wahono, 2003). *Sequence diagram* digunakan untuk menggambarkan skenario atau langkah-langkah yang dilakukan sebagai respon dari sebuah event yang menghasilkan *output* tertentu. Berikut adalah simbol-simbol dari *sequence diagram*:

Tabel 2.9 Simbol-Simbol *Sequence Diagram*

No	Nama	Gambar	Fungsi
1	<i>Actor</i>		Menggambarkan orang yang berinteraksi dengan sistem
2	<i>Boundary</i>		Menggambarkan antarmuka dan interaksi satu atau lebih aktor dengan sistem
3	<i>Controller</i>		Menggambarkan perilaku mengatur, mengkoordinasikan perilaku sistem dan dinamika dari suatu sistem, menangani tugas utama dan mengontrol alur kerja suatu sistem
4	<i>Entity</i>		Menggambarkan informasi atau data yang harus disimpan oleh sistem
5	<i>Lifeline</i>		Mengindikasikan keberadaan sebuah obyek dalam basis waktu
6	<i>Message</i>		Menggambarkan pesan / hubungan antar objek, yang menunjukkan urutan kejadian yang terjadi

Sumber: A.S & Shalahuddin [2011]

2.7.4 Class Diagram

Class diagram digunakan untuk memodelkan tampilan desain statis suatu sistem. *Class* menggambarkan sebuah keadaan (atribut/properti) suatu sistem, sekaligus memberikan layanan untuk memanipulasi keadaan tersebut. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain (Dharwiyanti & Wahono, 2003).

Class memiliki tiga area pokok :

1. Nama (dan stereotype)
2. Atribut
3. Metoda (metoda/fungsi)

Atribut dan metoda dapat memiliki salah satu sifat berikut :

- a. *Private*, tidak dapat dipanggil dari luar class yang bersangkutan

- b. *Protected*, hanya dapat dipanggil oleh class yang bersangkutan dan anak-anak yang mewarisinya
- c. *Public*, dapat dipanggil oleh siapa saja

Hubungan antar kelas:

1. Asosiasi, yaitu hubungan statis antar class. Umumnya menggambarkan class yang memiliki atribut berupa class lain, atau class yang harus mengetahui eksistensi class lain. Panah navigability menunjukkan arah query antar class.
2. Agregasi, yaitu hubungan yang menyatakan bagian ("terdiri atas..").
3. Pewarisan, yaitu hubungan hirarkis antar class. Class dapat diturunkan dari class lain dan mewarisi semua atribut dan metoda class asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari class yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.
4. Hubungan dinamis, yaitu rangkaian pesan (message) yang di-passing dari satu class kepada class lain. Hubungan dinamis dapat digambarkan dengan menggunakan sequence diagram yang akan dijelaskan kemudian.

2.8 Physical Data Model

Pemodelan data dalam PDM dapat dilakukan berdasarkan hasil pemodelan kelas dalam *class diagram*. Pemodelan PDM dapat dilakukan dengan 3 strategi, di antaranya yaitu (Rumbaugh, Jacobson, & Booch, 1998):

1. Mendefinisikan masing kelas pada *class diagram* menjadi tabel pada PDM. Hal ini merupakan strategi yang paling sederhana, namun akan mendatangkan kesulitan ketika rancangan *class diagram* mengalami perubahan, misal, terdapat *subclass* yang ditambahkan.
2. Mengumpulkan *subclass* dengan *superclass* pada tingkat hirarki yang sama, kemudian definisikan masing-masing kelas menjadi tabel pada PDM. Kekurangan strategi ini adalah akan menyebabkan kapasitas penyimpanan cepat berkurang.
3. Memisahkan *subclass* dengan *superclass* dan mendefinisikannya menjadi tabel pada PDM. Kekurangan dari strategi ini adalah akan membutuhkan mekanisme *join table* untuk proses pengambilan data melalui basis data.

2.9 CodeIgniter

CodeIgniter (CI) adalah sebuah *framework* yang digunakan untuk membantu dalam pembuatan sebuah sistem berbasis web yang disusun dengan menggunakan bahasa PHP. Kemudahan dalam penggunaan serta beberapa macam kelas yang berbentuk *library* dan *helper* juga sangat membantu programmer dalam mengembangkan aplikasinya.

Menurut Riyanto (Riyanto, 2011) *framework* merupakan sekumpulan perintah-perintah dasar atau fungsi yang dikumpulkan dalam satu kesatuan agar memudahkan pengembang untuk dipakai dalam pengembangan selanjutnya. Setiap *framework* memiliki aturan dan perintah yang berbeda berdasarkan fungsi yang ada di dalamnya.

2.9.1 Model, View dan Controller pada CI

CodeIgniter merupakan salah satu *framework* PHP yang menerapkan konsep *Model, View, dan Controller* (MVC). MVC merupakan sebuah pola perancangan yang memungkinkan pengembang memisahkan kode program mereka ke dalam tiga bagian, yakni *Model, View, dan Controller*. Ketiga bagian tersebut saling berhubungan satu sama lain. Berikut penjelasan mengenai konsep MVC:

1. Model

Model merupakan bagian didesain untuk bekerja melalui database. *Model* diartikan juga sebagai pembuat *prototype* data. Pada bagian biasanya berisi berbagai *query* yang ditujukan pada sebuah database. *Query* tersebut disesuaikan dengan kebutuhan data yang akan diolah oleh *controller*, yang nantinya akan ditampilkan menggunakan *view*.

2. View

View berfungsi sebagai penampil *user interface*. Di dalam bagian *view* akan diletakkan file yang berhubungan dengan tampilan akhir kepada *user*.

3. Controller

Controller merupakan bagian yang mengatur hubungan antara bagian model dan bagian view. *Controller* berfungsi untuk menerima *request* dan data dari user kemudian menentukan apa yang akan diproses oleh aplikasi.

2.10 Pengujian Perangkat Lunak

Tahap pengujian perangkat lunak memiliki tujuan untuk memeriksa fungsionalitas, kinerja dan kesesuaian dari struktur program yang telah dibuat. Setiap unit dari program-program yang sudah ada kemudian diuji sebagai satu kesatuan untuk memastikan apakah kebutuhan sistem sudah sesuai dan terpenuhi (Sommerville, 2011). 2 tujuan spesifik dalam proses pengujian perangkat lunak, yaitu:

1. Untuk memastikan bahwa perangkat lunak sudah memenuhi dan sesuai dengan seluruh persyaratan
2. Untuk mencari kemungkinan kesalahan yang terjadi pada sistem atau hal-hal yang tidak sesuai dengan spesifikasi.

1.10.1 Validation Testing

Pengujian validasi memiliki tujuan untuk mengetahui apakah sistem yang dibangun sudah sesuai dan benar dengan kebutuhan yang sudah ditetapkan

diawal. Fungsi-fungsi yang telah dijabarkan dalam daftar kebutuhan yang merupakan hasil dari analisis kebutuhan akan dijadikan acuan untuk melakukan pengujian validasi. Pengujian validasi dengan menggunakan metode pengujian *Black Box*, karena tidak memerlukan konsentrasi terhadap alur jalannya algoritma program dan lebih menekankan untuk menemukan konformitas antara kinerja sistem dengan daftar kebutuhan (Indriati, 2010).

2.10.2 User Acceptance Testing

User Acceptance testing adalah tahap akhir sebelum sistem diserahkan sepenuhnya kepada perusahaan untuk digunakan. Sistem akan diuji oleh pengguna sistem untuk mengetahui jika sistem atau perangkat lunak telah berjalan dengan baik dan sesuai dengan spesifikasi kebutuhan sistem pada fase penggalan kebutuhan sistem (Sommerville, 2011).

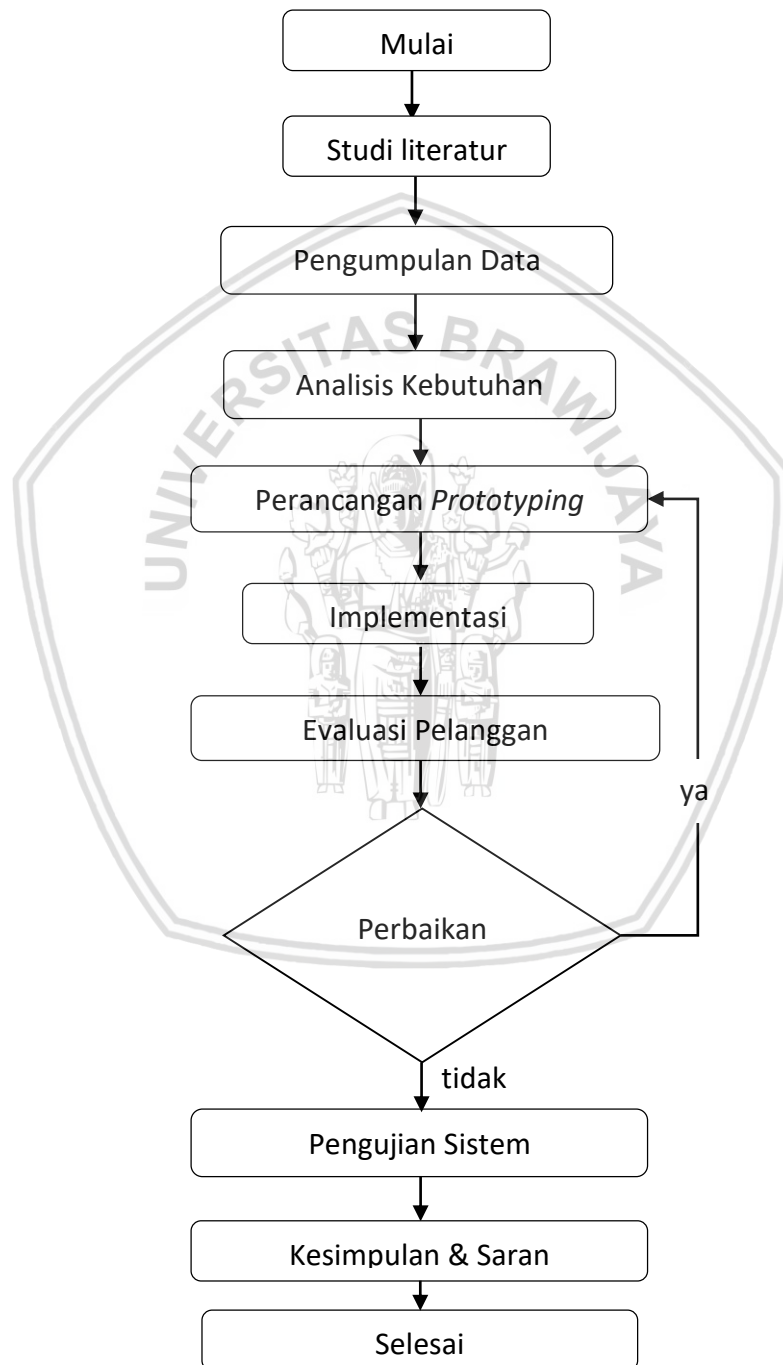
2.10.3 Compatibility Testing

Pengujian kompatibilitas merupakan salah satu dari beberapa pengujian yang perlu dilakukan pada antarmuka pengguna aplikasi berbasis web untuk memastikan apakah aplikasi tidak memiliki masalah ketika dijalankan pada komputer, perangkat tampilan, sistem operasi, aplikasi peramban, atau kecepatan jaringan yang berbeda (Pressman, 2010). Pengujian kompatibilitas pada aplikasi berbasis web dilakukan dengan memberikan definisi pada sekumpulan konfigurasi komputasi pada perangkat klien dan menguji aplikasi pada konfigurasi tersebut.



BAB 3 METODOLOGI

Pada bab ini menjelaskan tahapan yang harus ditentukan terlebih dahulu. Dengan adanya metode penelitian penyelesaian masalah akan memiliki alur yang terarah dan sistematis. Tahap-tahap pada metodologi penelitian yang dilakukan adalah studi literatur, analisis dan pengumpulan data, analisis kebutuhan, perancangan *prototyping*, implementasi, pengujian & evaluasi dan kesimpulan & saran.



Gambar 3.1 Diagram Alur Penelitian

3.1 Studi Literatur

Pada tahap ini dilakukan untuk merealisasikan tujuan dan penyelesaian masalah dengan mempelajari dan mengambil berbagai referensi yang didapat melalui buku, jurnal, skripsi, laporan penelitian, maupun artikel. Referensi mengenai *Customer Relationship Management* dan *Sales Force Automation* akan menjadi bagian utama dalam penulisan skripsi ini.

3.2 Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data dengan tujuan memahami proses bisnis yang sedang berjalan dan kegiatan yang terjadi di lingkungan perusahaan. Pengumpulan data yang diperlukan menggunakan 3 metode pengumpulan, berikut metode pengumpulan data:

- a. Observasi
Metode pengumpulan data ini dengan melakukan pengamatan kegiatan secara langsung di Branded IT Store.
- b. Wawancara
Metode pengumpulan data dilakukan dengan wawancara terhadap pihak yang terkait mengenai permasalahan yang ada di Branded IT Store
- c. Studi Literatur Ulang
Pada tahap ini dilakukan dengan memahami pustaka secara detail. Hal ini dilakukan bertujuan untuk mendapatkan gambaran yang lebih jelas mengenai sistem yang akan dibangun berdasarkan permasalahan yang terjadi di Branded IT Store.

Berdasarkan pengumpulan data yang telah dilakukan melalui wawancara yang terlampir pada lampiran A, didapat permasalahan yang muncul di Branded IT Store adalah belum mempunyai sistem yang dapat mengelola data pelanggan dan penjualan, hal ini dapat memungkinkan terjadinya kesalahpahaman antara penjual dan pelanggan.

3.3 Analisis Kebutuhan

Pada tahap ini bertujuan untuk memperoleh kebutuhan yang diperlukan oleh pengguna sistem pada Branded IT Store. Langkah-langkah yang dilakukan mulai dari wawancara secara langsung kepada *Manager* Branded IT Store, kemudian dilanjutkan dengan melakukan survey dan analisis terkait proses bisnis yang berjalan pada Branded IT Store. Kemudian menentukan gambaran umum sistem, menentukan aktor yang terkait dengan sistem, analisa data yang akan dikelola dan disimpan dan melakukan pemodelan *use case diagram* dan *activity diagram*.

3.4 Perancangan

Pada tahap ini menjelaskan tentang perancangan dari sistem yang akan dibuat. Hasil dari tahap ini digunakan sebagai bahan referensi untuk mengimplementasikan sistem yang akan dibuat. Pada perancangan ini dilakukan

pembuatan *sequence diagram*, pembuatan *class diagram* dan perancangan *database*.

3.5 Implementasi

Pada tahap ini penulis mengimplementasikan perancangan prototyping yang sudah dilakukan pada tahap sebelumnya untuk memenuhi kebutuhan sistem. Pembuatan prototyping ini menggunakan bahasa pemrograman PHP dengan memanfaatkan framework CI.

3.6 Evaluasi Pelanggan

Pada tahap ini menjelaskan bagaimana pelanggan melakukan evaluasi. Evaluasi pelanggan dilakukan dengan mendemonstrasikan sistem informasi kepada pelanggan. Hal tersebut dibutuhkan untuk mengetahui apakah rancangan sistem informasi sudah dapat diterima atau belum. Jika sistem informasi yang dibuat sudah sesuai dengan kebutuhan yang telah ditetapkan diawal maka tahap selanjutnya dapat dilakukan, jika sistem informasi masih belum sesuai dengan kebutuhan yang telah dibuat, maka akan kembali ke tahap perancangan *prototyping*.

3.7 Pengujian

Pada tahap pengujian, hasil dari *prototyping* akan diuji. Pengujian *prototyping* dilakukan bertujuan untuk memeriksa kesesuaian kebutuhan pelanggan dengan *prototyping* yang sudah dibuat. Pengujian sistem ini, bertujuan untuk mengetahui apakah sistem yang telah dibuat berjalan sesuai dengan keinginan atau tidak, sehingga dapat diketahui kekurangan atau kelemahan dari sistem agar dapat dilakukan perbaikan. Dalam penelitian ini penulis menggunakan pengujian *black-box*.

Pengujian *black-box* dalam penelitian ini menggunakan *validation testing* yang merupakan pengujian terhadap fungsional suatu perangkat lunak dimana peneliti menyiapkan satu situasi masukan valid atau tidak valid, kemudian diperiksa keluaran dari hasil fungsi tersebut apakah sudah sesuai dengan yang diharapkan atau tidak. Sebelum sistem diserahkan sepenuhnya kepada perusahaan, dilakukan pengujian terhadap sistem oleh manager atau yang bertanggung jawab terhadap sistem yang dibangun dengan *user acceptance testing* (UAT). Untuk mengetahui komptibilitas pada sistem yang dibangun maka dilakukan pengujian kompatibilitas dengan menggunakan aplikasi *SortSite*.

3.8 Kesimpulan & Saran

Pada tahap ini, kesimpulan dan saran dapat diperoleh melalui penilaian mengenai keseluruhan perangkat lunak yang telah dibuat dan diuji. Sehingga sistem informasi penjualan yang telah dibuat dapat memenuhi kebutuhan pengguna dan menyelesaikan masalah yang telah dirumuskan pada latar belakang.

BAB 4 ANALISIS KEBUTUHAN

Bab ini menjelaskan tentang analisis kebutuhan sistem informasi *Sales Force Automation* pada Branded IT Store Malang. Analisis kebutuhan merupakan tahapan pertama yang harus dilakukan sebelum melakukan perancangan dan implementasi perangkat lunak. Pada penelitian ini analisis kebutuhan diawali dengan deskripsi umum perangkat lunak, analisis proses bisnis, identifikasi aktor yang terlibat dalam sistem, spesifikasi kebutuhan dan pemodelan *use case diagram* dan *activity diagram*. Tahapan tersebut bertujuan untuk memberikan gambaran kebutuhan apa saja yang akan dimiliki oleh sistem.

4.1 Deskripsi Umum Perangkat Lunak

Sistem informasi ini memiliki beberapa fungsi yang didapatkan dari hasil analisis wawancara terhadap Branded IT Store Malang yang menghasilkan kebutuhan perangkat lunak yang akan dijadikan sebagai dasar dalam pembuatan sistem. Kebutuhan perangkat lunak tersebut dijelaskan pada tabel 4.1.

Tabel 4.1 Deskripsi umum perangkat lunak

No.	Tipe Kebutuhan	Nama Fitur
1	Fungsional	Sistem menyediakan antarmuka untuk mengelola produk yang dimiliki.
2	Fungsional	Sistem menyediakan antarmuka untuk mengelola penjualan
3	Fungsional	Sistem menyediakan antarmuka untuk mengelola data pelanggan
4	Fungsional	Sistem menyediakan antarmuka khusus untuk <i>Manager</i> mengelola pegawai.
5	Fungsional	Sistem menyediakan antarmuka khusus untuk <i>Manager</i> melihat laporan penjualan
6	Fungsional	Sistem menyediakan antarmuka khusus untuk <i>Manager</i> melihat pencapaian pegawai.
7	Fungsional	Sistem menyediakan antarmuka khusus untuk <i>Manager</i> melihat laporan keluar masuk barang.
8	Fungsional	Sistem menyediakan antarmuka untuk melihat produk bagi pelanggan.
9	Fungsional	Sistem menyediakan antarmuka untuk melakukan pembelian produk bagi pelanggan.
10	Non-Fungsional	Sistem dapat diakses pada berbagai macam <i>web-browser</i>

4.2 Analisis Proses Bisnis

Pada tahap analisis proses bisnis bertujuan untuk mencari masalah yang perlu diselesaikan dan mengetahui pemangku kepentingan dari sistem. Langkah pertama yang akan dilakukan yaitu dengan melakukan wawancara secara langsung kepada pengelola toko, kemudian langkah selanjutnya adalah melakukan analisis terhadap proses bisnis yang telah ada sebelumnya (*as-is*). Berdasarkan hasil analisis terhadap proses bisnis *as-is* akan didapatkan usulan proses bisnis yang akan diterapkan (*to-be*) yang ditawarkan oleh peneliti kepada pemangku kepentingan. Proses bisnis *to-be* akan digunakan untuk menggambarkan perubahan serta tambahan yang akan diusulkan pada sistem.

Analisis proses bisnis dilakukan dengan menggunakan BPMN untuk menggambarkan proses bisnis *as-is* dan *to-be* pada proses penjualan di Branded IT Store. Analisis proses bisnis bertujuan untuk mengetahui alur proses penjualan dan identifikasi aktor yang akan dirancang.

4.2.1 Proses Bisnis As-Is

4.2.1.1 Proses Bisnis Pembelian dari Supplier As-Is

Pada gambar 4.1 proses pembelian melalui supplier terjadi serangkaian aktivitas untuk memenuhi kebutuhan dari perusahaan. Proses bisnis yang terjadi pada saat ini dimulai ketika perusahaan melakukan pemesanan barang ke supplier. Supplier akan memberikan informasi terkait dengan barang yang dipesan tersedia atau tidak. Jika barang tersedia maka perusahaan akan melanjutkan pemesanan. Setelah perusahaan melanjutkan pemesanan, perusahaan melakukan pembayaran dan barang yang dipesan akan dikirim oleh supplier. Ketika barang tidak tersedia maka perusahaan dapat memilih untuk tetap melanjutkan pemesanan dengan menunggu hingga barang tersedia atau membatalkan pemesanan yang telah dilakukan. Pencatatan masih dilakukan secara manual di mana perusahaan memasukkan data barang yang masuk ke dalam file excel.

Dari analisis proses bisnis pembelian dari supplier *as-is* dapat ditemukan beberapa kelemahan dan kekurangan berupa:

1. Proses pencatatan barang yang masuk masih manual belum ada sistem database yang menampung catatan barang masuk.

4.2.1.2 Proses Bisnis Penjualan As-Is

Proses bisnis mencakup proses awal penjualan di toko hingga akhir proses penjualan. Saat ini Branded IT Store masih melakukan pencatatan secara manual terkait dengan penjualan yang dilakukan oleh pelanggan. Proses transaksi yang dilakukan secara *offline*, membuat konsumen yang ingin membeli produk di Branded IT Store harus datang langsung ke toko Branded IT Store. Perusahaan juga masih belum memiliki database pelanggan yang dapat dikelola. Seharusnya jika perusahaan memiliki database pelanggan, perusahaan dapat mengetahui

mengetahui karakteristik pelanggannya. Berikut adalah proses bisnis penjualan *As-Is* yang dijelaskan pada gambar 4.2.

Dari analisis proses bisnis penjualan *as-is* dapat ditemukan beberapa kelemahan dan kekurangan berupa :

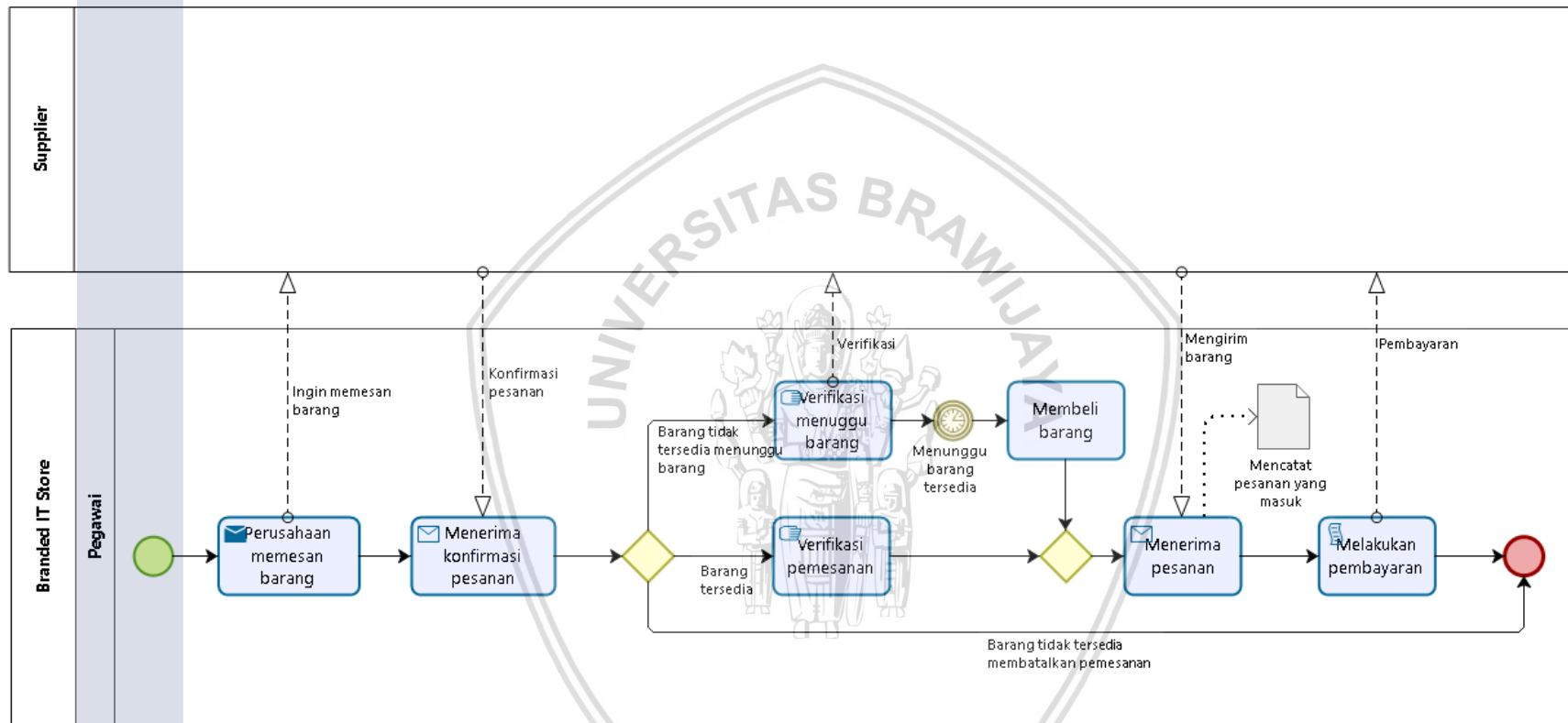
1. Pelayanan yang diberikan oleh Branded IT Store masih bersifat konvensional yaitu perusahaan akan memberikan informasi produk ketika pelanggan (customer) datang ke toko.
2. Belum adanya sarana tempat bertukar informasi antara Branded IT Store dengan para pelanggan secara Real-time.
3. Perusahaan tidak mengetahui karakteristik dari pelanggan yang telah membeli produknya.

4.2.1.3 Proses Bisnis Pelunasan *As-Is*

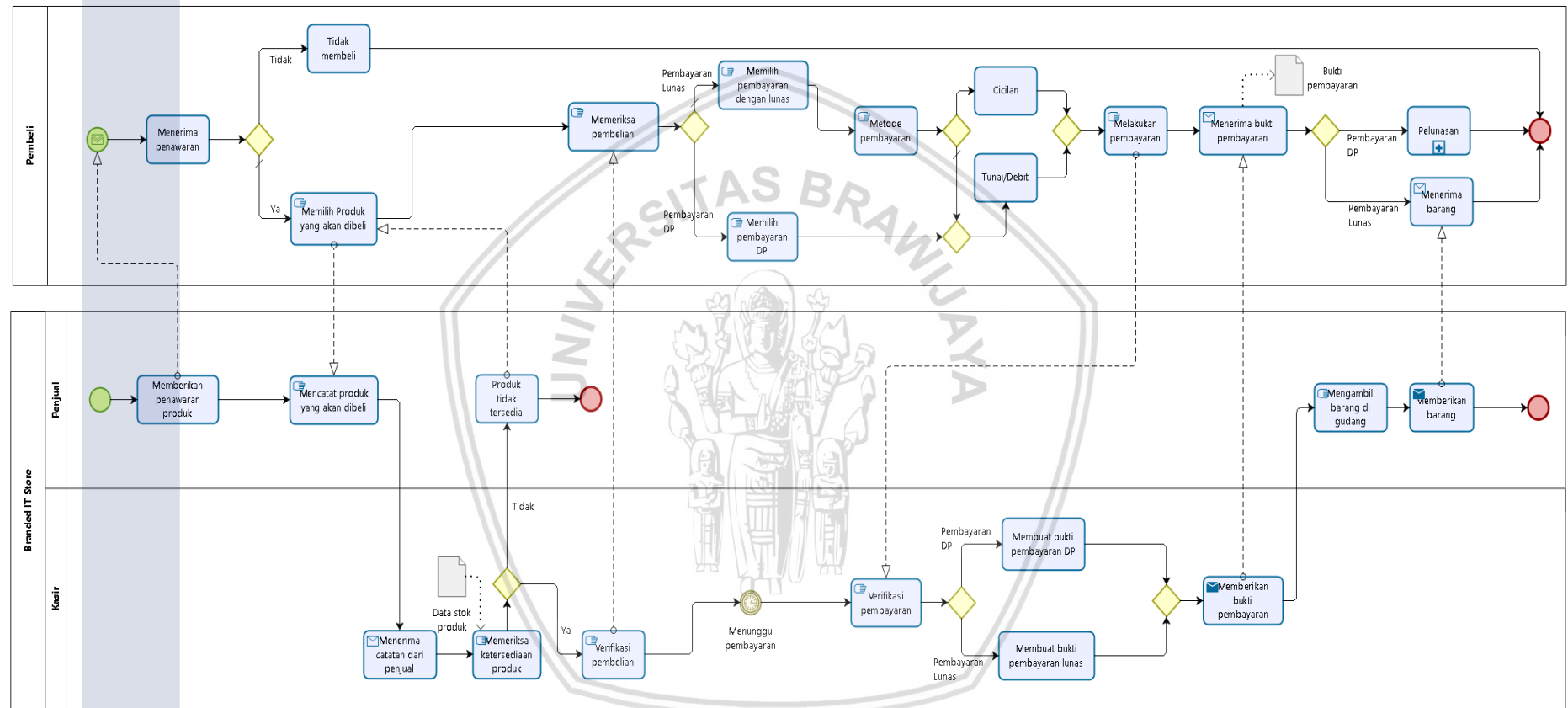
Saat ini pada bagian pelunasan di Branded IT Store masih menggunakan cara manual yaitu pelanggan yang sudah menerima kertas berupa bukti pembayaran DP pada proses bisnis penjualan memberikan bukti tersebut kepada Branded IT Store. Branded IT Store belum mempunyai data terkait dengan pembayaran DP yang sudah terjadi pada sebelumnya. Proses bisnis berlanjut setelah pelanggan memberikan bukti pembayaran DP dan melakukan kekurangan pembayaran yang ditagihkan. Setelah pelunasan selesai Branded IT Store akan memproses pembayaran dan menyiapkan barang yang sudah dilunasi oleh pelanggan. Berikut adalah proses bisnis pelunasan *As-Is* yang dijelaskan pada gambar 4.3.

Dari analisis proses bisnis pelunasan *as-is* dapat ditemukan beberapa kelemahan dan kekurangan berupa:

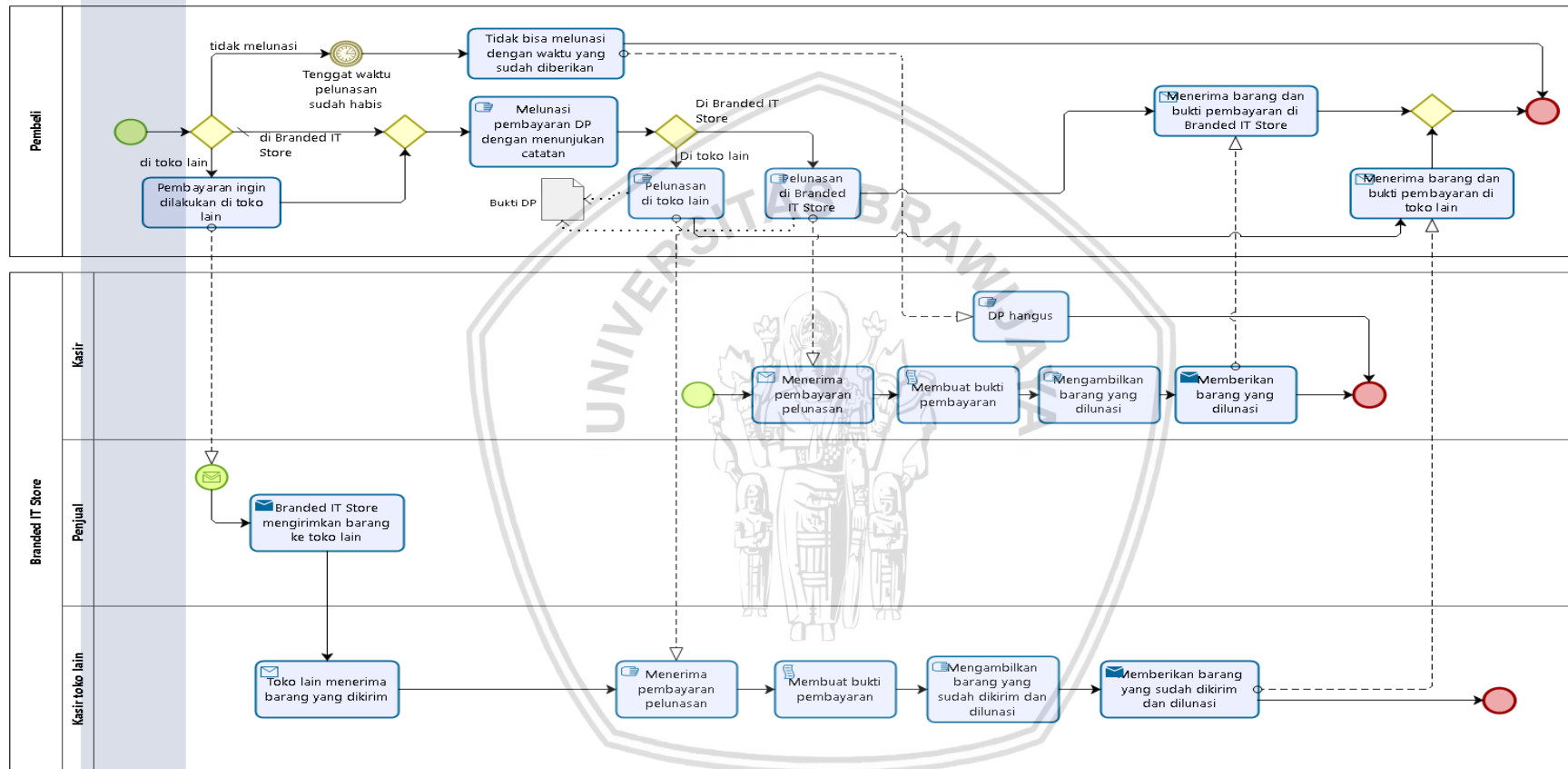
1. Proses transaksi pada bagian pelunasan yang diberikan oleh Branded IT Store masih secara manual dan hanya dimiliki pelanggan.



Gambar 4.1 Proses Bisnis Pembelian dari Supplier As-Is



Gambar 4.2 Proses Bisnis Penjualan As-Is



Gambar 4.3 Proses Bisnis Pelunasan As-Is

4.2.2 Proses Bisnis To-Be

4.2.2.1 Proses Bisnis Pembelian dari Supplier *To-Be*

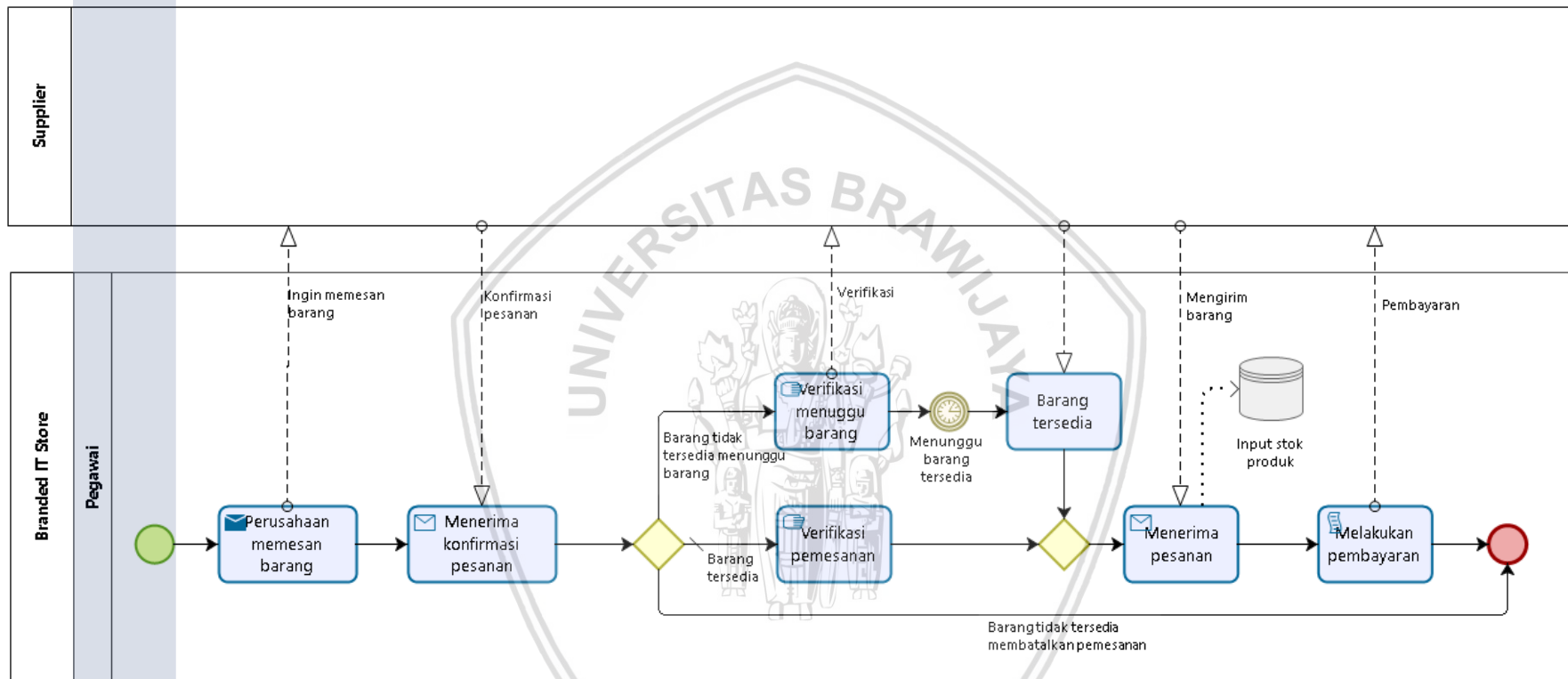
Pada gambar 4.4 menunjukkan proses bisnis pembelian dari supplier *to-be* yang dilakukan oleh Branded IT Store. Branded IT Store yang mendapatkan kemudahan untuk mengelola penyimpanan data setelah toko melakukan pembelian. Data pembelian yang dilakukan oleh toko disimpan didalam data produk. Proses bisnis yang berubah hanya pada bagian penyimpanan data yang sudah menggunakan *database*.

4.2.2.2 Proses Bisnis Penjualan *To-Be*

Pada gambar 4.5 menunjukkan proses penjualan yang dilakukan oleh pelanggan yang akan digunakan oleh perusahaan. Para pelanggan mendapatkan kemudahan untuk melakukan pembelian melalui web kapan pun tanpa perlu datang terlebih dahulu ke toko. Pada website Branded IT Store juga di tampilkan informasi terkait produk-produk yang tersedia. Proses bisnis penjualan terjadi ketika pelanggan mengunjungi website atau datang langsung ke toko. Pelanggan dapat memilih barang sesuai dengan katalog dan stok barang yang ada. Setelah pelanggan memilih barang, pelanggan dapat melakukan pembayaran dan akan diproses oleh perusahaan. Pelanggan dapat menerima langsung barang yang dibeli jika melakukan pembayaran lunas atau cicilan, pada pilihan pembayaran DP pelanggan harus melunasi barang terlebih dahulu.

4.2.2.3 Proses Bisnis Pelunasan *To-Be*

Pada gambar 4.6 menunjukkan bagaimana perusahaan dapat mengetahui terkait pelunasan pembayaran yang akan dilakukan oleh pelangg. Pelangga juga dapat melakukan pembayaran melalui web yang disediakan. Pelanggan hanya perlu memasukan nomer pembayaran DP yang didapat setelah pelanggan melakukan pembelian pada proses bisnis penjualan. Setelah memasukan nomer pembayaran, pelanggan harus mengunggah bukti pembayaran sisa yang sudah dibayarkan. Ketika pelanggan tidak menyelesaikan pelunasan dari waktu yang sudah diberikan, maka pembayaran DP yang sudah dilakukan sebelumnya akan hangus.



Gambar 4.4 Proses Bisnis Pembelian dari Supplier To-Be



4.3 Identifikasi Aktor

Identifikasi aktor dilakukan untuk mengetahui sebuah peran yang dilakukan oleh manusia atau sistem lainnya yang terlibat dengan sistem yang ada secara langsung. Pada tahap identifikasi aktor, peneliti mengacu pada tahap analisis kebutuhan yang telah dilakukan sebelumnya, pada tabel 4.2 menjelaskan aktor dan deskripsi aktor yang berhubungan dengan sistem

Tabel 4.2 Identifikasi Aktor

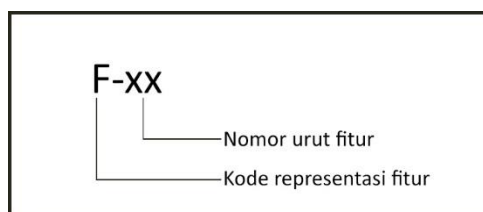
No	Aktor	Deskripsi
1	Pengguna	Pengguna merupakan sebuah peran yang dapat menggunakan sistem dan melakukan login sesuai dengan hak akses.
2	Manager	Manager merupakan sebuah peran yang bertanggung jawab atas perusahaan, dapat menggunakan sistem untuk <i>monitoring</i> penjualan, mengelola data produk dan mengelola data pelanggan serta pegawai yang ada di Branded IT Store.
3	Penjual	Penjual merupakan sebuah peran yang dapat menggunakan sistem untuk mengelola penjualan, data pelanggan dan data produk.
4	Kasir	Kasir merupakan sebuah peran yang dapat menggunakan sistem untuk mengelola penjualan.
5	Pelanggan	Pelanggan merupakan sebuah peran yang dapat menggunakan sistem untuk melihat produk dan melakukan pembelian produk.

4.4 Spesifikasi Kebutuhan Sistem

Pada spesifikasi kebutuhan terdiri dari kebutuhan fungsional dan non-fungsional dari sistem yang akan dibuat.

4.4.1 Fitur Sistem

Kebutuhan sistem didapat melalui hasil wawancara dengan *stakeholder* perusahaan yang terlampir pada lampiran A. Pada tabel 4.3 menunjukkan fitur sistem berdasarkan fungsionalitas yang dapat dilakukan oleh aktor. Fitur-fitur pada penelitian ini memiliki kode sebagai identitas. Keterangan kodifikasi fitur terdapat dalam gambar 4.7



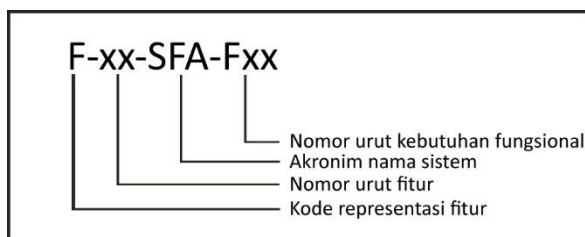
Gambar 4.7 Kodifikasi Fitur

Fitur yang terdapat dalam penelitian ini merepresentasikan solusi yang diberikan untuk menyelesaikan masalah dan dapat memenuhi kebutuhan pengguna. Pada tabel 4.3 menjelaskan fitur sistem dari hasil identifikasi fitur yang disediakan untuk dapat memenuhi kebutuhan pengguna.

Tabel 4.3 Fitur Sistem

Kode Fitur	Deskripsi	Nama Fitur
F-01	Sistem menyediakan antarmuka untuk mengelola produk yang dimiliki.	Mengelola Data Produk
F-02	Sistem menyediakan antarmuka untuk mengelola penjualan	Mengelola Data Penjualan
F-03	Sistem menyediakan antarmuka untuk mengelola data pelanggan	Mengelola Data Pelanggan
F-04	Sistem menyediakan antarmuka khusus untuk <i>Manager</i> mengelola pegawai.	Mengelola Data Pegawai
F-05	Sistem menyediakan antarmuka khusus untuk <i>Manager</i> melihat laporan penjualan	Melihat Laporan Penjualan
F-06	Sistem menyediakan antarmuka khusus untuk <i>Manager</i> untuk melihat pencapaian pegawai.	Melihat Pencapaian Pegawai
F-07	Sistem menyediakan antarmuka khusus untuk <i>Manager</i> untuk melihat laporan keluar masuk barang.	Melihat Laporan Keluar Masuk Barang
F-08	Sistem menyediakan antarmuka untuk melihat produk bagi pelanggan.	Melihat Produk
F-09	Sistem menyediakan antarmuka untuk melakukan pembelian produk bagi pelanggan.	Membeli Produk
F-10	Sistem menyediakan antarmuka untuk aktor agar dapat masuk dalam sistem.	Login

4.4.2 Kebutuhan Fungsional



Gambar 4.8 Kodifikasi Kebutuhan Fungsional

Kebutuhan fungsional ini menjelaskan mengenai informasi kemampuan sistem *Sales Force Automation* yang akan dibangun. Informasi berikut merupakan kebutuhan fungsional produk yang ditunjukkan pada tabel 4.5.

Tabel 4.4 Kebutuhan Fungsional

No	Kode Fitur	Kode Lengkap Fungsi	Deskripsi
1	F-01	F-01-SFA-F01	Sistem menyediakan fungsi untuk menambah data produk
		F-01-SFA-F02	Sistem menyediakan fungsi untuk memperbarui data produk
		F-01-SFA-F03	Sistem menyediakan fungsi untuk menampilkan daftar data produk
2	F-02	F-02-SFA-F01	Sistem menyediakan fungsi untuk menambah data penjualan
		F-02-SFA-F02	Sistem menyediakan fungsi untuk memperbarui data penjualan
		F-02-SFA-F03	Sistem menyediakan fungsi untuk menampilkan daftar data penjualan
3	F-03	F-03-SFA-F01	Sistem menyediakan fungsi untuk menambah data pelanggan
		F-03-SFA-F02	Sistem menyediakan fungsi untuk memperbarui data pelanggan
		F-03-SFA-F03	Sistem menyediakan fungsi untuk menampilkan daftar data pelanggan
4	F-04	F-04-SFA-F01	Sistem menyediakan fungsi untuk menambah data pegawai
		F-04-SFA-F02	Sistem menyediakan fungsi untuk memperbarui pegawai
		F-04-SFA-F03	Sistem menyediakan fungsi untuk menghapus daftar data pegawai

Tabel 4.4 Kebutuhan Fungsional (lanjutan)

No	Kode Fitur	Kode Lengkap Fungsi	Deskripsi
		F-04-SFA-F04	Sistem menyediakan fungsi untuk menampilkan daftar data pegawai
5	F-05	F-05-SFA-F01	Sistem menyediakan fungsi untuk melihat laporan penjualan
		F-05-SFA-F02	Sistem menyediakan fungsi untuk mencetak laporan penjualan
		F-05-SFA-F03	Sistem menyediakan fungsi untuk mengexport data laporan penjualan kedalam excel
6	F-06	F-06-SFA-F01	Sistem menyediakan fungsi untuk menampilkan pencapaian pegawai
7	F-07	F-07-SFA-F01	Sistem menyediakan fungsi untuk menampilkan laporan barang keluar masuk
8	F-08	F-08-SFA-F01	Sistem menyediakan fungsi untuk melihat produk-produk
9	F-09	F-09-SFA-F01	Sistem menyediakan fungsi untuk melihat produk
10	F-10	F-10-SFA-F01	Sistem menyediakan fungsi untuk melakukan pembelian produk

4.4.3 Kebutuhan Non-Fungsional

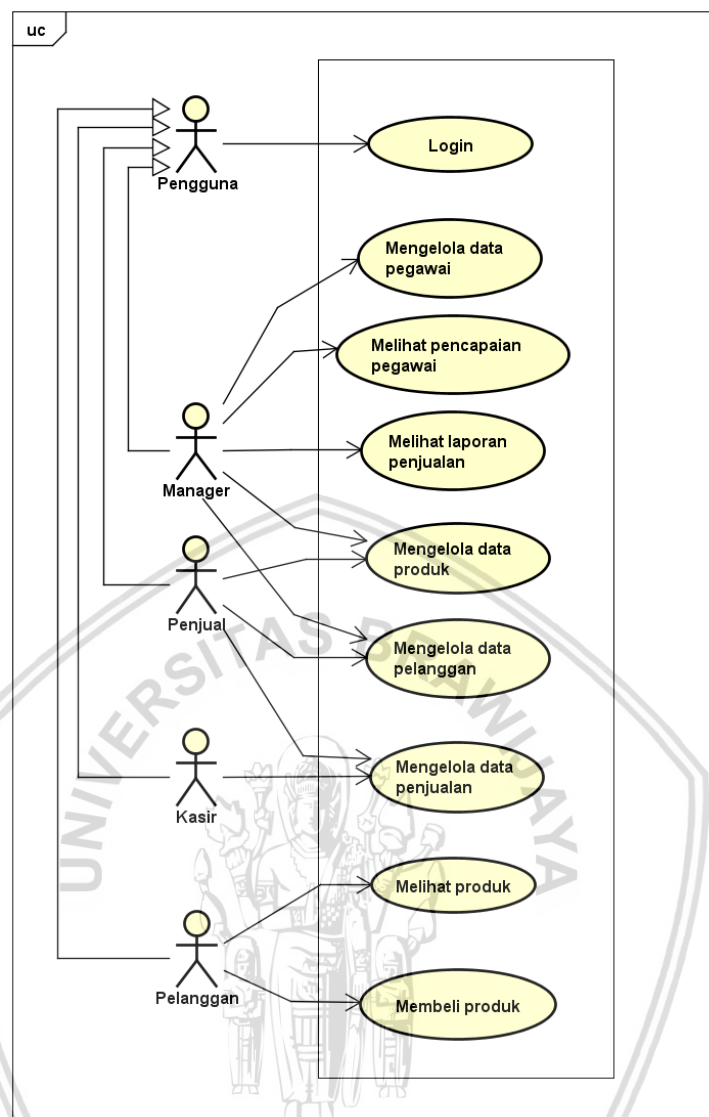
Kebutuhan non-fungsional didapat dari hasil identifikasi masalah yang telah dilakukan, berikut merupakan kebutuhan non fungsional sistem berdasarkan fungsionalitas yang dapat dilakukan oleh aktor pada tabel 4.4.

Tabel 4.5 Kebutuhan Non-Fungsional

No	Kode Fungsi	Deskripsi
1	F-11-SFA-NF01	Sistem dapat dibuka pada berbagai macam web browser

4.5 Pemodelan Use- Case Diagram

Use case diagram adalah sebuah diagram UML yang mewakili aktor-aktor yang terlibat pada sistem dan peran yang akan dijalankan, pada use case diagram ini didapatkan melalui kebutuhan fungsionalitas sistem yang sudah dijabarkan sebelumnya.



Gambar 4.9 Use Case Diagram

4.5.1 Spesifikasi Use Case

Tiap *use case* yang sudah didefinisikan sebelumnya dijelaskan secara detail menggunakan *use case specification*. *Use case specification* dijelaskan dalam bentuk tabel yang berisi nama *use case*, tujuan *use case*, aktor yang terlibat, spesifikasi utama, dan spesifikasi alternatif. Hasil spesifikasi *use case* merupakan penjabaran dari *use case diagram* pada gambar 4.9.

Use case yang sudah teridentifikasi, selanjutnya dihubungkan dengan solusi yang ditawarkan dalam aktivitas proses bisnis *to-be*. Penghubungan yang dilakukan antara *use case* dengan beberapa aktivitas proses bisnis *to-be* ditujukan untuk dapat mengetahui hubungan yang lebih baik dalam penggunaan pemodelan proses bisnis dan *use case* dalam proses pengembangan sistem informasi.

Tabel 4.6 Pemetaan Proses Bisnis *To-Be* dengan *Use Case*

Proses Bisnis	Aktivitas	Use Case
Pembelian dari Supplier	Menerima pesanan	Mengelola data produk
Penjualan	Melakukan registrasi	Mengelola data pelanggan
	Memasukan data pelanggan	
	Mengunggah bukti pembayaran	Membeli barang
	Memilih produk yang ada di katalog sistem	
	Memberikan produk yang sudah dibeli	Mengelola data produk
	Memilih produk	Mengelola data penjualan
	Memperbarui status penjualan	
	Melakukan pembayaran	
Pelunasan	Melunasi sisa pembayaran di toko	Mengelola data produk
	Status produk yang dibeli lunas	
	Memberikan produk yang dibeli	

Berikut ini merupakan spesifikasi dari masing-masing use case yang telah digambarkan sebelumnya pada gambar 4.9, berikut merupakan spesifikasi *usecase*:

4.5.1.1 Spesifikasi *use case Login*

Pada tabel 4.6, akan menjelaskan spesifikasi *use case Login*.

Tabel 4.7 Spesifikasi *Use Case Login*

<i>Flow of Events</i> untuk <i>use case login</i>	
Brief Description	<i>Use case</i> ini menjelaskan bagaimana <i>Manager</i> , Kasir dan Penjual masuk ke dalam Sistem Informasi <i>Sales Force Automation</i>
Actor	<i>Manager</i> , Kasir dan Penjual
Pre-Condition	Mengakses Sistem Informasi <i>Sales Force Automation</i>
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor berada dalam halaman login 2. Aktor memasukkan <i>username</i> dan <i>password</i>, lalu mengakses fungsi login 3. Sistem melakukan pengecekan bagi Aktor yang akan masuk ke dalam sistem

Tabel 4.7 Spesifikasi *Use Case Login* (lanjutan)

	4. Sistem menampilkan halaman beranda jika <i>username</i> dan <i>password</i> valid oleh sistem {use case selesai}
Alternative Flows	<ul style="list-style-type: none"> {username dan password} tidak valid <p>Jika, di <i>basic flow</i>, Aktor memasukkan <i>username</i> dan <i>password</i> tidak valid, sistem menampilkan pesan kesalahan. Aktor bisa memilih untuk kembali ke awal basic flow atau membatalkan login, dan <i>use case</i> selesai</p>
Subflow	Tidak ada
Post-Conditions	Aktor berhasil mengakses fungsi login dan dapat masuk ke dalam sistem

4.5.1.2 Spesifikasi *use case* mengelola pegawai

Pada tabel 4.7, akan menjelaskan spesifikasi *use case* mengelola pegawai.

Tabel 4.8 Spesifikasi *Use Case Mengelola Pegawai*

Flow of Events untuk use case mengelola pegawai	
Brief Description	<i>Use case</i> ini menjelaskan bagaimana <i>Manager</i> melakukan pengelolaan data pegawai meliputi penambahan, perbarui dan menghapus data pegawai
Actor	<i>Manager</i>
Pre-Condition	Status <i>login</i> aktor sukses
Basic Flow of Events	<ol style="list-style-type: none"> Aktor memilih fungsi untuk menampilkan data pegawai Sistem menampilkan data pegawai dan fungsi perbarui dan hapus pada <i>tab</i> daftar data pegawai dan sistem juga menampilkan <i>tab</i> tambah data pegawai untuk menambah data pegawai Jika Aktor memilih <i>tab</i> tambah data pegawai, maka sistem akan menjalankan <i>sub flow</i> tambah data pegawai Jika Aktor memilih fungsi ubah data pegawai pada <i>tab</i> data pegawai, maka sistem akan menjalankan <i>sub flow</i> perbarui data pegawai Jika Aktor memilih fungsi hapus data pegawai pada <i>tab</i> data pegawai, maka sistem akan menjalankan <i>sub flow</i> hapus data pegawai {use case selesai}

Tabel 4.8 Spesifikasi *Use Case* Mengelola Pegawai (lanjutan)

Alternative Flows	<ul style="list-style-type: none"> • Data yang dimasukkan tidak lengkap <p>Pada {memilih fungsi tambah data pegawai} di <i>basic flow</i> dan {mengisi form data pegawai} pada <i>sub flow</i>, jika inputan aktor masih kosong maka sistem akan membaca inputan masih kosong dan menampilkan pesan bahwa terdapat data yang belum terisi.</p>
Subflow	<p>Tambah data pegawai</p> <ol style="list-style-type: none"> 1. Aktor memilih <i>tab</i> tambah data pegawai 2. Sistem akan menampilkan <i>form</i> untuk menambah data pegawai 3. Aktor mengisi <i>form</i> dan memasukkan data pegawai lalu memilih fungsi simpan 4. Sistem berhasil menambahkan data pegawai <p>Perbarui data pegawai</p> <ol style="list-style-type: none"> 1. Aktor memilih fungsi perbarui pada tab daftar data pegawai 2. Sistem menampilkan <i>form</i> untuk memperbarui data pegawai 3. Aktor memperbarui data pegawai dan memilih fungsi simpan 4. Sistem berhasil merubah data pegawai <p>Hapus data pegawai</p> <ol style="list-style-type: none"> 1. Aktor memilih fungsi hapus pada <i>tab</i> daftar data pegawai 2. Sistem melakukan operasi dan verifikasi penghapusan data pegawai 3. Aktor mengkonfirmasi hapus data 4. Sistem berhasil menghapus data pegawai
Post-Conditions	Aktor berhasil mengelola data pegawai

4.5.1.3 Spesifikasi *use case* melihat pencapaian pegawai

Pada tabel 4.8, akan menjelaskan spesifikasi *use case* melihat pencapaian pegawai.

Tabel 4.9 Spesifikasi *Use Case* Melihat Pencapaian Pegawai

Flow of Events untuk <i>use case</i> pencapaian pegawai	
Brief Description	<i>Use case</i> ini menjelaskan bagaimana <i>Manager</i> dapat melihat pencapaian pegawai
Actor	<i>Manager</i>

Tabel 4.9 Spesifikasi *Use Case* Melihat Pencapaian Pegawai (lanjutan)

Pre-Condition	Status <i>login</i> aktor sukses
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih fungsi pencapaian pegawai 2. Sistem menampilkan data pencapaian pegawai dan fungsi detail pada <i>tab</i> daftar data pegawai. 3. Jika Aktor memilih fungsi detail pencapaian pegawai, maka sistem akan menjalankan <i>sub flow detail pencapaian {use case selesai}</i>
Alternative Flows	-
Subflow	Detail penjualan <ol style="list-style-type: none"> 1. Aktor memilih fungsi detail pencapaian 2. Sistem menampilkan halaman detail pencapaian yang dipilih
Post-Conditions	Aktor berhasil melihat data pencapaian pegawai

4.5.1.4 Spesifikasi *use case* melihat laporan penjualan

Pada tabel 4.9, akan menjelaskan spesifikasi *use case* melihat laporan penjualan.

Tabel 4.10 Spesifikasi *Use Case* Melihat Laporan Penjualan

Flow of Events untuk <i>use case</i> laporan penjualan	
Brief Description	<i>Use case</i> ini menjelaskan bagaimana <i>Manager</i> dapat melihat laporan penjualan secara detail
Actor	<i>Manager</i>
Pre-Condition	Status <i>login</i> aktor sukses
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih fungsi laporan penjualan 2. Sistem menampilkan data laporan penjualan dan fungsi detail pada <i>tab</i> daftar data penjualan. 3. Jika Aktor memilih fungsi detail laporan penjualan, maka sistem akan menjalankan <i>sub flow detail laporan penjualan {use case selesai}</i>
Alternative Flows	-
Subflow	Detail laporan penjualan <ol style="list-style-type: none"> 1. Aktor memilih fungsi detail penjualan 2. Sistem menampilkan halaman detail penjualan yang dipilih
Post-Conditions	Aktor berhasil melihat laporan penjualan

4.5.1.5 Spesifikasi *use case* mengelola data produk

Pada tabel 4.11, akan menjelaskan spesifikasi *use case* mengelola data produk.

Tabel 4.11 Spesifikasi *Use Case* Mengelola Data Produk

Flow of Events untuk <i>use case</i> mengelola data produk	
Brief Description	<i>Use case</i> ini menjelaskan bagaimana <i>Manager, Kasir</i> dan <i>Penjual</i> mengelola data produk meliputi tambah, hapus dan perbarui data produk
Actor	<i>Manager</i> dan <i>Penjual</i>
Pre-Condition	Status <i>login</i> aktor sukses
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih fungsi data produk 2. Sistem menampilkan data produk serta fungsi tambah sn, perbarui dan hapus data produk pada <i>tab</i> daftar data produk dan sistem juga menampilkan <i>tab</i> tambah data produk untuk menambah produk. 3. Jika Aktor memilih fungsi tambah data produk, maka sistem akan menjalankan subflow tambah data produk 4. Jika Aktor memilih fungsi perbarui data produk, maka sistem akan menjalankan subflow perbarui data produk 5. Jika Aktor memilih fungsi tambah sn, maka sistem akan menjalankan subflow tambah sn
Alternative Flows	<ul style="list-style-type: none"> • Data yang dimasukkan tidak lengkap <p>Pada {memilih fungsi tambah data produk} di <i>basic flow</i> dan {mengisi form data produk} pada <i>sub flow</i>, jika inputan aktor masih kosong maka sistem akan membaca inputan masih kosong dan menampilkan pesan bahwa terdapat data yang belum terisi.</p>
Subflow	<p>Tambah data produk</p> <ol style="list-style-type: none"> 1. Aktor memilih fungsi tambah produk 2. Sistem akan menampilkan <i>form</i> untuk menambah produk 3. Aktor mengisi <i>form</i> dan memasukkan data produk lalu memilih fungsi simpan 4. Sistem menampilkan pesan bahwa data berhasil ditambahkan <p>Perbarui data produk</p> <ol style="list-style-type: none"> 1. Aktor memilih memilih fungsi perbarui data produk pada halaman daftar data pelanggan.

Tabel 4.11 Spesifikasi *Use Case* Mengelola Data Produk (lanjutan)

	<ol style="list-style-type: none"> 2. Sistem akan menampilkan <i>form</i> untuk memperbarui pelanggan. 3. Aktor mengisi <i>form</i> dan memperbarui data pelanggan lalu pilih fungsi simpan. 4. Sistem menampilkan pesan bahwa data berhasil diperbarui. <p>Tambah sn</p> <ol style="list-style-type: none"> 1. Aktor memilih fungsi tambah sn 2. Sistem akan menampilkan <i>form</i> untuk menambah sn dari produk yang dipilih 3. Aktor mengisi <i>form</i> dan memasukkan data sn lalu pilih simpan 4. Sistem menyimpan masukan data dan data berhasil ditambahkan
Post-Conditions	Aktor berhasil mengelola data produk

4.5.1.6 Spesifikasi *use case* mengelola data pelanggan

Pada tabel 4.12, akan menjelaskan bagaimana spesifikasi *use case* mengelola data pelanggan.

Tabel 4.12 Spesifikasi *Use Case* Mengelola Data Pelanggan

Flow of Events untuk <i>use case</i> mengelola data pelanggan	
Brief Description	<i>Use case</i> ini menjelaskan bagaimana <i>Manager</i> dan penjual mengelola data pelanggan meliputi tambah, hapus dan perbarui data pelanggan
Actor	<i>Manager</i> dan Penjual
Pre-Condition	Status <i>login</i> aktor sukses
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih fungsi data pelanggan 2. Sistem menampilkan data pelanggan dan fungsi perbarui dan hapus pada <i>tab</i> daftar data pelanggan dan sistem menampilkan <i>tab</i> tambah data pelanggan untuk menambah data pelanggan 3. Jika Aktor memilih fungsi tambah data pelanggan, maka sistem akan menjalankan <i>sub flow tambah data pelanggan</i> 4. Jika Aktor memilih fungsi ubah data pelanggan, maka sistem akan menjalankan <i>sub flow perbarui data pelanggan</i> {use case selesai}
Alternative Flows	<ul style="list-style-type: none"> • Data yang dimasukkan tidak lengkap

Tabel 4.12 Spesifikasi *Use Case* Mengelola Data Pelanggan(lanjutan)

	Pada {memilih fungsi tambah data pelanggan} di <i>basic flow</i> dan {mengisi form data pelanggan} pada <i>sub flow</i> , jika inputan aktor masih kosong maka sistem akan membaca inputan masih kosong dan menampilkan pesan bahwa terdapat data yang belum terisi.
Subflow	<p>Tambah pelanggan</p> <ol style="list-style-type: none"> 1. Aktor memilih halaman tambah pelanggan 2. Sistem akan menampilkan <i>form</i> untuk menambah pelanggan 3. Aktor mengisi <i>form</i> dan memasukkan data pelanggan lalu pilih fungsi simpan 4. Sistem menampilkan pesan bahwa data berhasil ditambahkan <p>Perbarui pelanggan</p> <ol style="list-style-type: none"> 1. Aktor memilih fungsi perbarui pelanggan pada <i>tab</i> daftar data pelanggan 2. Sistem akan menampilkan <i>form</i> untuk memperbarui pelanggan 3. Aktor mengisi <i>form</i> dan memperbarui data pelanggan lalu pilih simpan 4. Sistem menampilkan pesan bahwa data berhasil diperbarui
Post-Conditions	Aktor berhasil mengelola data pelanggan

4.5.1.7 Spesifikasi *use case* mengelola data penjualan

Pada tabel 4.13, akan menjelaskan bagaimana spesifikasi *use case* mengelola data penjualan.

Tabel 4.13 Spesifikasi *Use Case* Mengelola Data Penjualan

Flow of Events untuk <i>use case</i> mengelola data penjualan	
Brief Description	<i>Use case</i> ini menjelaskan bagaimana Kasir dan Penjual mengelola data penjualan meliputi tambah, perbarui dan hapus data penjualan
Actor	Kasir dan Penjual
Pre-Condition	Status <i>login</i> aktor sukses
Basic Flow of Events	<ol style="list-style-type: none"> 1. Aktor memilih fungsi data penjualan 2. Sistem menampilkan form pada <i>tab</i> tambah data penjualan dan sistem juga menampilkan <i>tab</i> data

Tabel 4.13 Spesifikasi *Use Case* Mengelola Data Penjualan (lanjutan)

	<p>penjualan serta fungsi perbarui dan hapus pada <i>tab</i> data penjualan</p> <ol style="list-style-type: none"> 3. Jika Aktor memilih fungsi tambah data penjualan, maka sistem akan menjalankan <i>sub flow</i> tambah data penjualan 4. Jika Aktor memilih fungsi perbarui data penjualan, maka sistem akan menjalankan <i>sub flow</i> perbarui data penjualan 5. Jika Aktor mengubah jumlah DP, maka sistem akan menjalankan <i>sub flow</i> ubah status pembayaran {use case selesai}
Alternative Flows	<ul style="list-style-type: none"> • Status pembayaran DP Pada {ubah status pembayaran} di <i>basic flow</i>, apabila aktor memasukkan jumlah pembayaran DP maka status pembayaran akan menjadi DP • Data yang dimasukkan tidak lengkap Pada {memilih fungsi tambah data penjualan} di <i>basic flow</i> dan {mengisi form data penjualan} pada <i>sub flow</i>, jika inputan aktor masih kosong maka sistem akan membaca inputan masih kosong dan menampilkan pesan bahwa terdapat data yang belum terisi.
Subflow	<p>Tambah penjualan</p> <ol style="list-style-type: none"> 1. Aktor memilih <i>tab</i> tambah data penjualan 2. Sistem akan menampilkan <i>form</i> untuk menambah data penjualan 3. Aktor mengisi <i>form</i> dan memasukkan data penjualan 4. Sistem menampilkan pesan bahwa data berhasil ditambahkan <p>Perbarui penjualan</p> <ol style="list-style-type: none"> 1. Aktor memilih fungsi perbarui data penjualan 2. Sistem akan menampilkan <i>form</i> untuk memperbarui data penjualan 3. Aktor mengisi <i>form</i> dan memperbarui data penjualan lalu pilih simpan 4. Sistem berhasil memperbarui data penjualan <p>Ubah status pembayaran</p>

Tabel 4.13 Spesifikasi *Use Case* Mengelola Data Penjualan(lanjutan)

	<ol style="list-style-type: none"> 1. Aktor memilih fungsi ubah penjualan pada <i>tab</i> data penjualan 2. Sistem menampilkan form perbarui data 3. Aktor mengubah pembayaran DP menjadi 0 4. Sistem mengubah status pembayaran menjadi lunas
Post-Conditions	Aktor berhasil mengelola penjualan

4.5.1.8 Spesifikasi *use case* melihat produk

Pada tabel 4.14, akan menjelaskan spesifikasi *use case* melihat produk yang dimiliki toko.

Tabel 4.14 Spesifikasi *Use Case* Melihat Produk

Flow of Events untuk <i>use case</i> melihat produk	
Brief Description	<i>Use case</i> ini menjelaskan bagaimana Pelanggan dapat melihat produk
Actor	Pelanggan
Pre-Condition	Pelanggan berada dalam menu beranda
Basic Flow of Events	<ol style="list-style-type: none"> 1. Sistem menampilkan data produk dan fungsi detail. 2. Jika Aktor memilih fungsi detail produk, maka sistem akan menjalankan <i>sub flow</i> detail produk {<i>use case</i> selesai}
Alternative Flows	-
Subflow	Detail produk <ol style="list-style-type: none"> 1. Aktor memilih fungsi detail produk 2. Sistem menampilkan halaman detail produk yang dipilih
Post-Conditions	Aktor berhasil melihat produk

4.5.1.9 Spesifikasi *use case* membeli produk

Pada tabel 4.15, akan menjelaskan spesifikasi *use case* membeli produk.

Tabel 4.15 Spesifikasi *Use Case* Membeli Produk

Flow of Events untuk <i>use case</i> membeli produk	
Brief Description	<i>Use case</i> ini menjelaskan bagaimana Pelanggan dapat membeli produk
Actor	Pelanggan
Pre-Condition	Status <i>login</i> aktor sukses

Tabel 4.15 Spesifikasi *Use Case* Membeli Produk(lanjutan)

Basic Flow of Events	<ol style="list-style-type: none"> 1. Sistem menampilkan data produk dan fungsi detail. 2. Jika Aktor memilih operasi beli, maka sistem akan menjalankan <i>sub flow</i> beli produk {<i>use case</i> selesai}
Alternative Flows	-
Subflow	Beli produk <ol style="list-style-type: none"> 1. Aktor memilih fungsi beli produk 2. Sistem menyimpan produk yang dibeli. 3. Aktor memilih fungsi <i>cart</i> untuk melihat detail pembelian. 4. Aktor memilih fungsi lanjutkan. 5. Sistem menampilkan form rincian. 6. Aktor memilih simpan. 7. Sistem berhasil menyimpan pembelian dan menunggu unggahan pembayaran. 8. Aktor mengunggah pembayaran. 9. Sistem menyimpan unggahan dan memproses pembayaran.
Post-Conditions	Aktor berhasil melakukan pembelian produk

4.6 Pemodelan *Activity Diagram*

Pemodelan *activity diagram* menggambarkan aliran aktivitas kerja pada sistem. Pemodelan *activity diagram* dilakukan untuk menjelaskan alur use case yang sebelumnya sudah dijelaskan agar lebih mudah dipahami oleh *stakeholder*. Berikut ini merupakan beberapa aktivitas yang digambarkan dalam penelitian yaitu, *activity diagram* login, *activity diagram* pengelolaan data pegawai, *activity diagram* pengelolaan data penjualan, dan *activity diagram* pengelolaan data produk pada sistem informasi Sales Force Automation.

1. *Activity diagram* login

Activity diagram login dimulai ketika pengguna mengakses sistem dan sistem menampilkan halaman login, kemudian pengguna mengisi username dan password, kemudian sistem akan mengecek masukan sebagai *Manager*, kasir atau penjual. Pengguna akan dapat masuk kedalam sistem jika data yang dimasukkan oleh pengguna benar, *activity diagram* login ditunjukkan pada gambar 4.10.

2. *Activity diagram* mengelola data pegawai

Activity diagram mengelola data pegawai dimulai pada saat *Manager* dapat mengakses ke dalam sistem dan berada dalam menu beranda, kemudian *Manager* memilih menu data pegawai. Lalu sistem menampilkan daftar data pegawai dan beberapa fungsi untuk melakukan pengelolaan data yang dapat

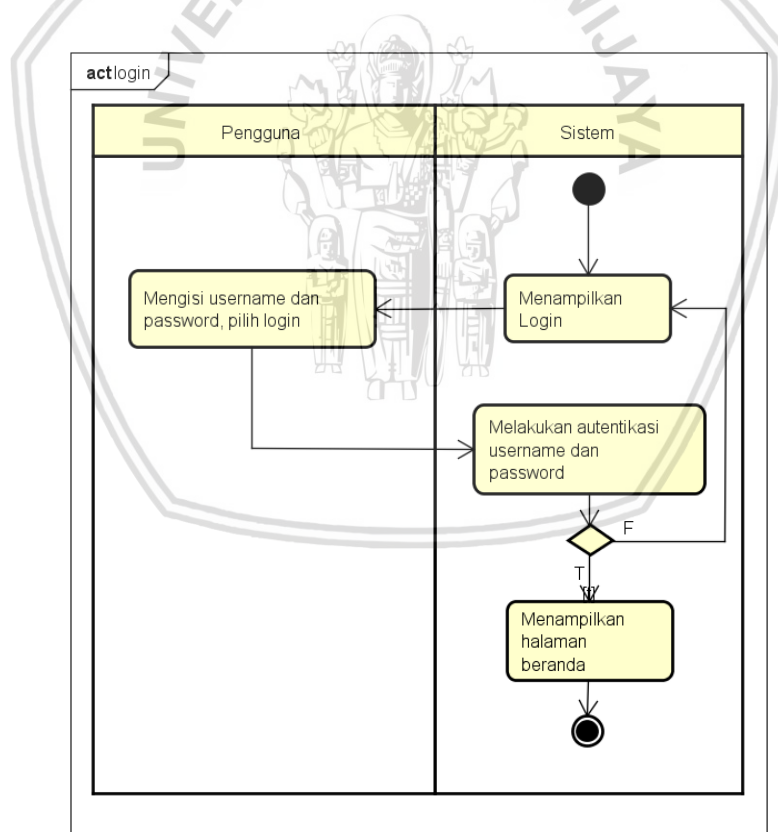
Manager pilih, *activity diagram* mengelola data pegawai ditunjukkan pada gambar 4.11.

3. *Activity diagram* mengelola data penjualan

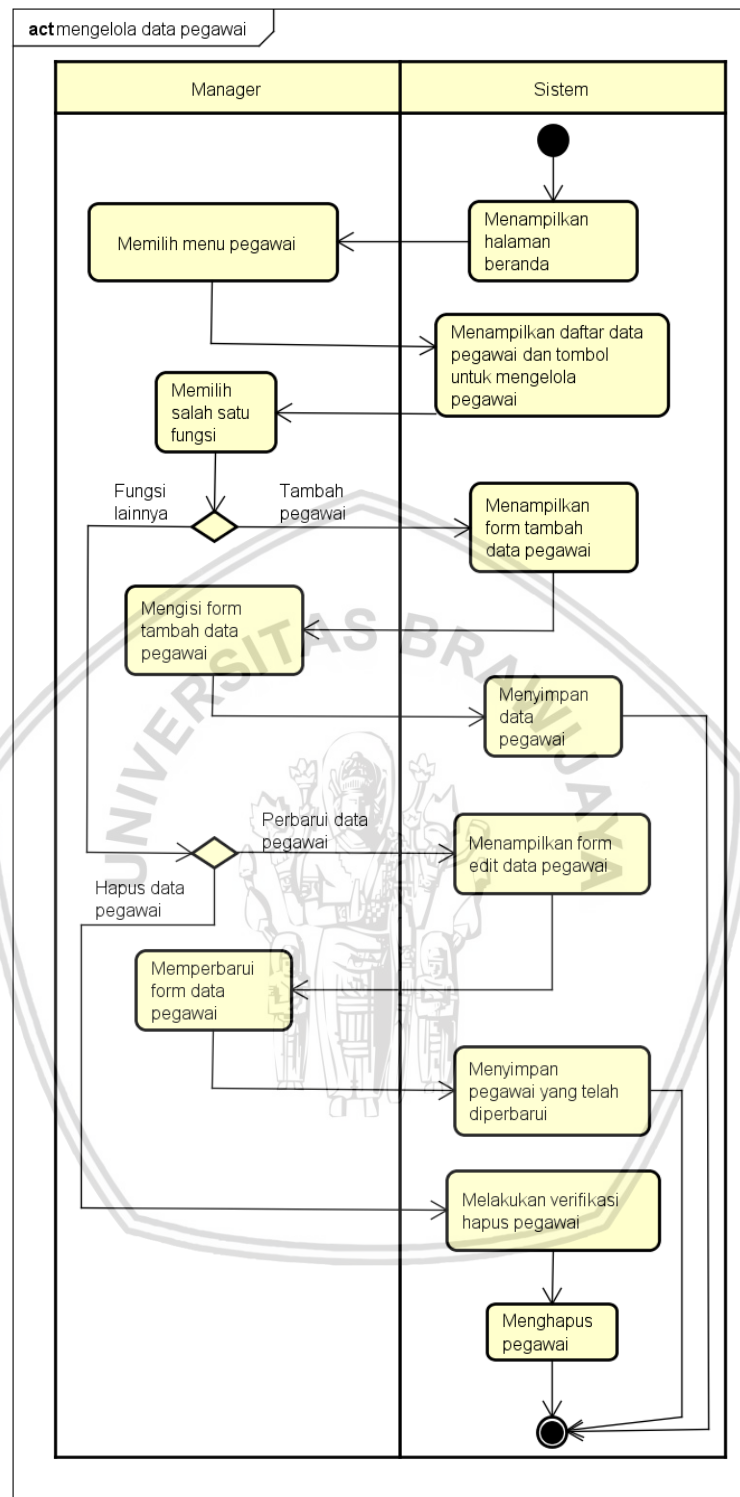
Activity diagram mengelola data penjualan dimulai pada saat kasir dan penjual dapat mengakses ke dalam sistem dan berada dalam menu data penjualan, lalu kasir dan penjual memilih menu data penjualan. Dalam menu data penjualan, sistem akan menampilkan halaman data penjualan yang berisi daftar data penjualan dan beberapa fungsi kelola yang dapat kasir dan penjual pilih, *activity diagram* mengelola data penjualan ditunjukkan pada gambar 4.12.

4. *Activity diagram* mengelola data produk

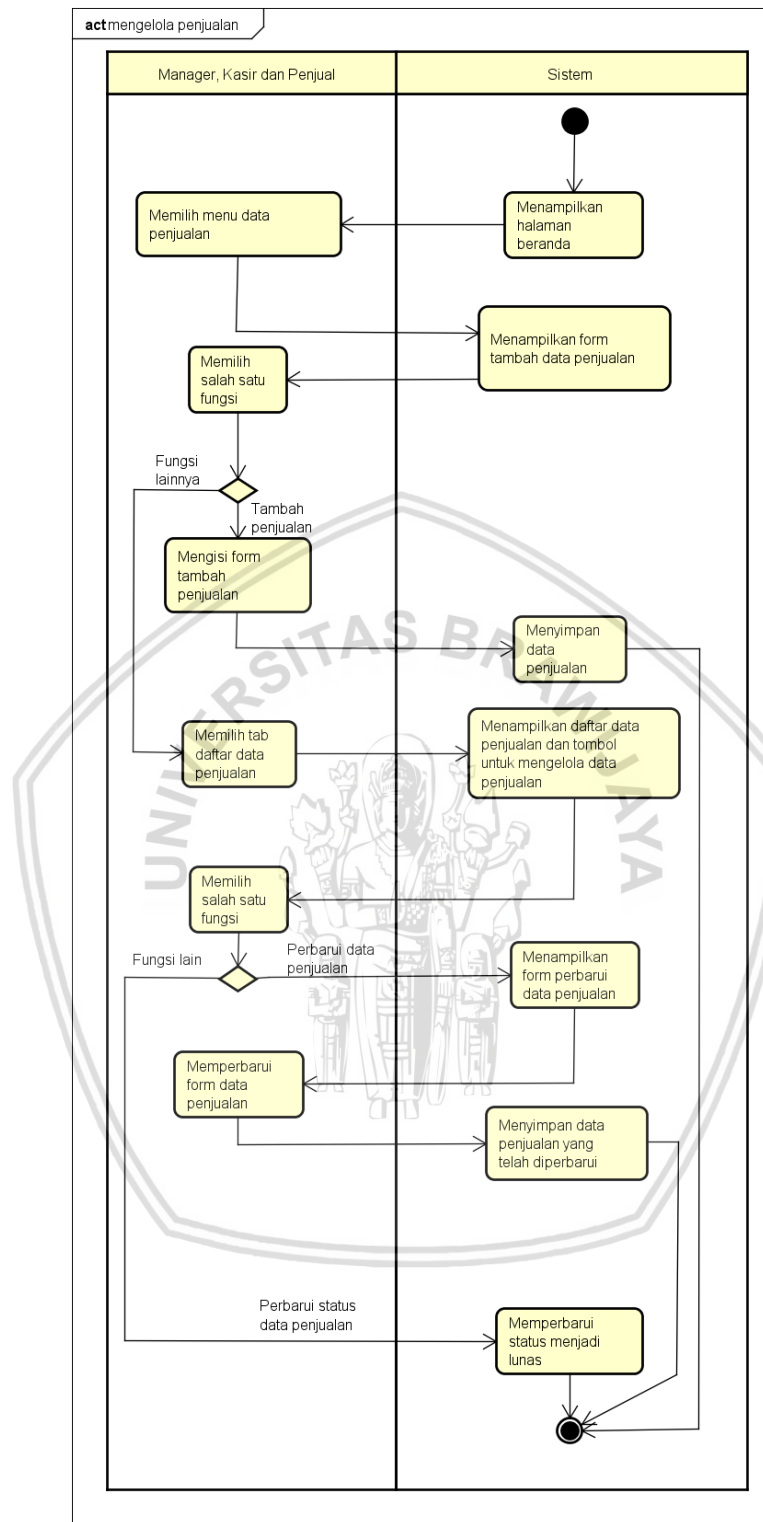
Activity diagram mengelola data produk dimulai pada saat *manager* dan penjual dapat mengakses ke dalam sistem dan berada dalam menu data produk, lalu *manager* dan penjual memilih menu data produk. Dalam menu data produk, sistem akan menampilkan halaman data produk yang berisi daftar data produk dan beberapa fungsi kelola yang dapat *manager* dan penjual pilih, *activity diagram* mengelola data penjualan ditunjukkan pada gambar 4.13.



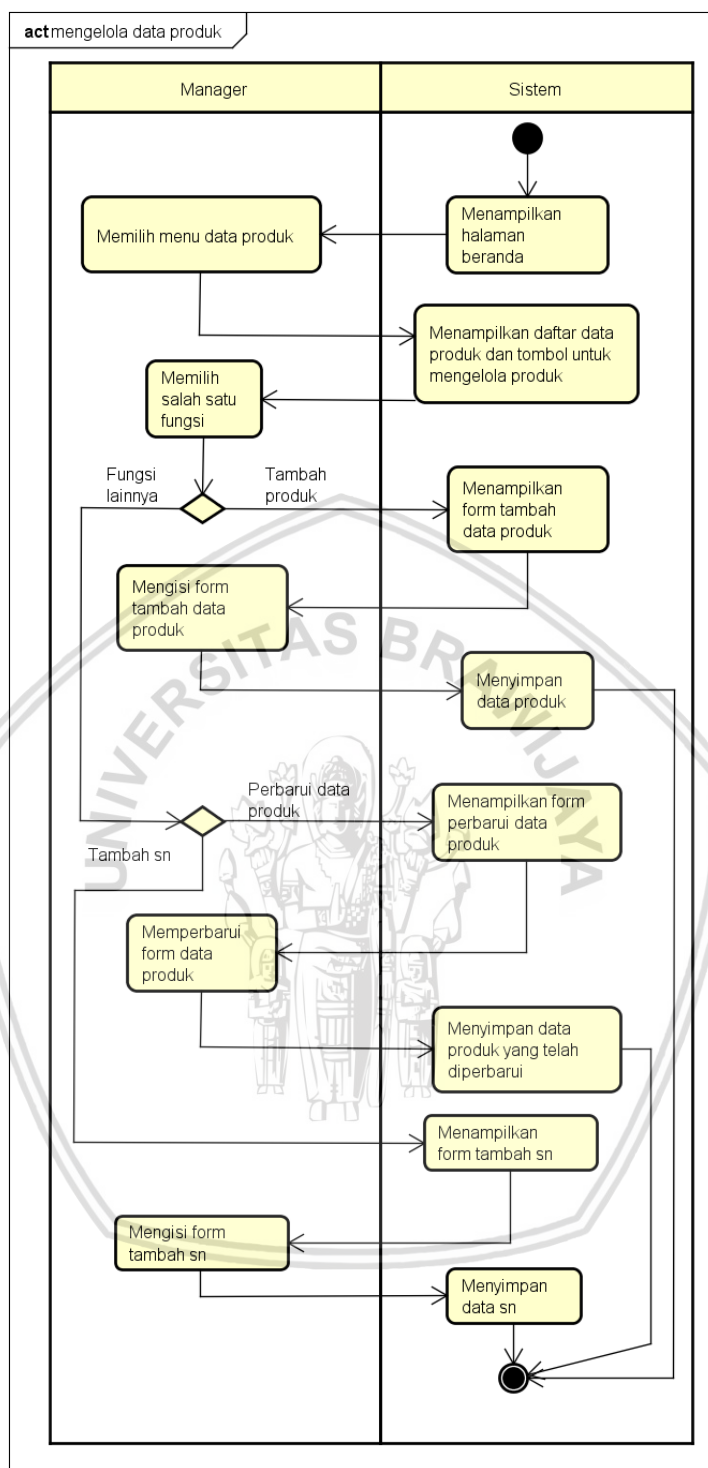
Gambar 4.10 Activity Diagram Login



Gambar 4.11 Activity Diagram Mengelola Data Pegawai



Gambar 4.12 Activity Diagram Mengelola Data Penjualan



Gambar 4.13 Activity Diagram Mengelola Data Produk

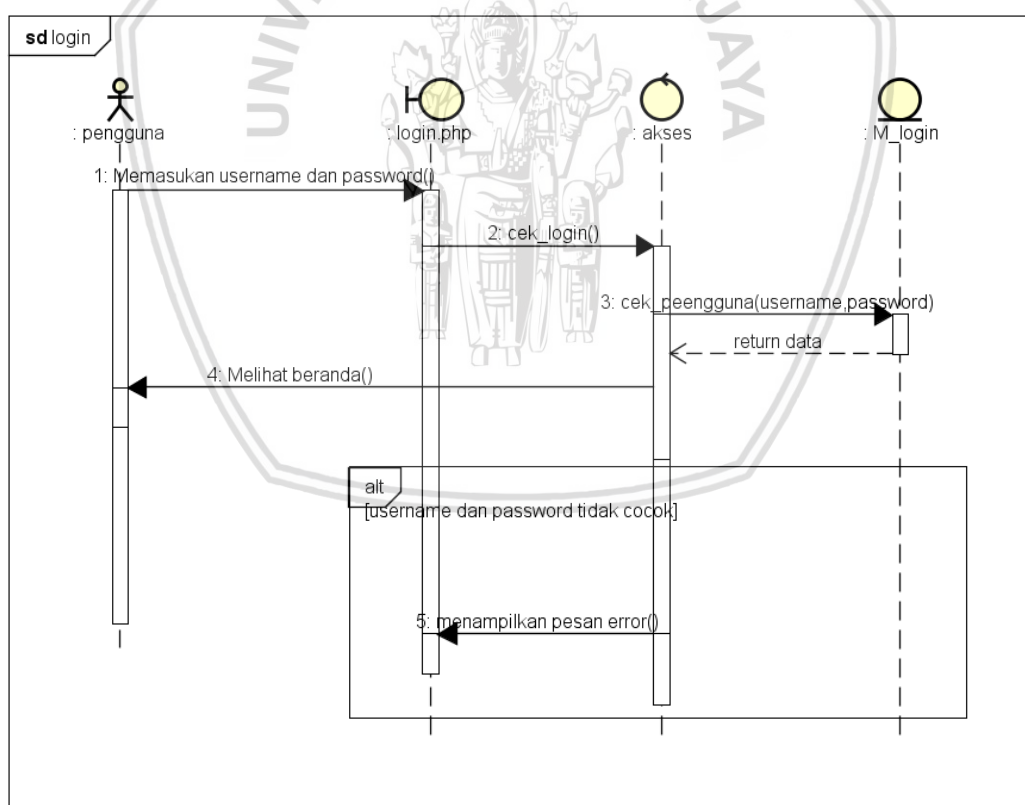
BAB 5 PERANCANGAN

5.1 Perancangan *Sequence Diagram*

Sequence Diagram memberikan gambaran sebuah objek yang berhubungan pada *use case* dengan memberikan penjelasan mengenai interaksi antar masing-masing objek dalam urutan waktu. Interaksi ini berupa pengiriman serangkaian data antar objek-objek yang saling berinteraksi.

1. *Sequence diagram login*

Sequence diagram login dimulai pada saat Pengguna melakukan akses ke sistem dan sistem menampilkan halaman *login*. Kemudian Pengguna memasukkan *username* dan *password*. Hasil input dari Pengguna akan dilanjutkan ke dalam controller login, dan dilanjutkan kembali ke dalam *M_login* untuk dilakukan pengecekan dalam basis data. Jika hasil pengecekan input sesuai maka akan menampilkan halaman beranda sistem sesuai dengan hak akses masing masing, jika hasil pengecekan tidak sesuai dengan input Pengguna maka akan kembali ke halaman *login* dan menampilkan pesan *error*. *Sequence diagram login* ditunjukkan pada gambar 5.1.

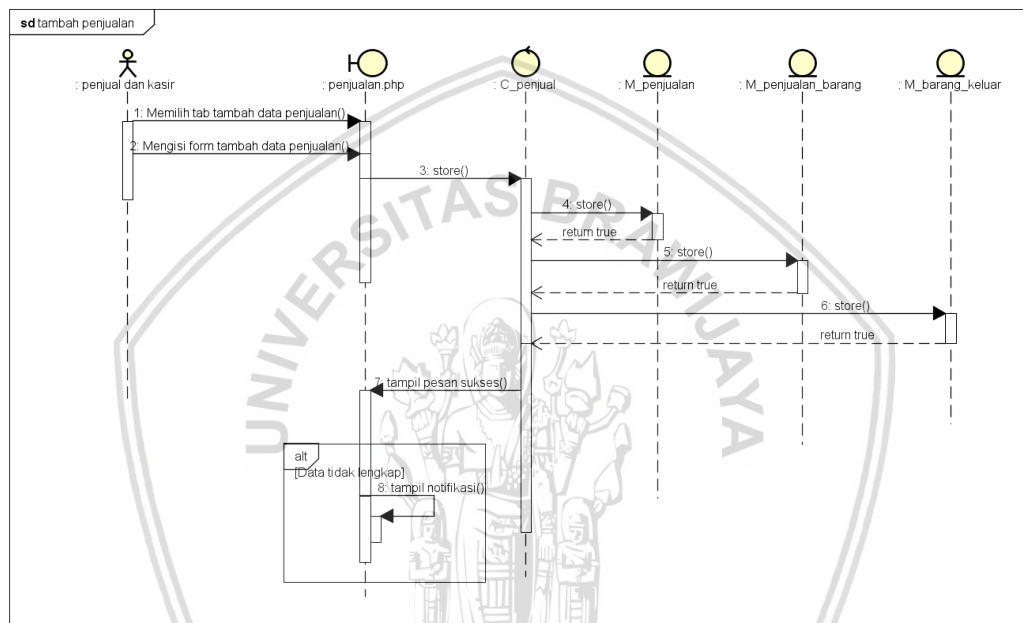


Gambar 5.1 Sequence Diagram Login

2. Sequence diagram mengelola penjualan

a. Tambah penjualan

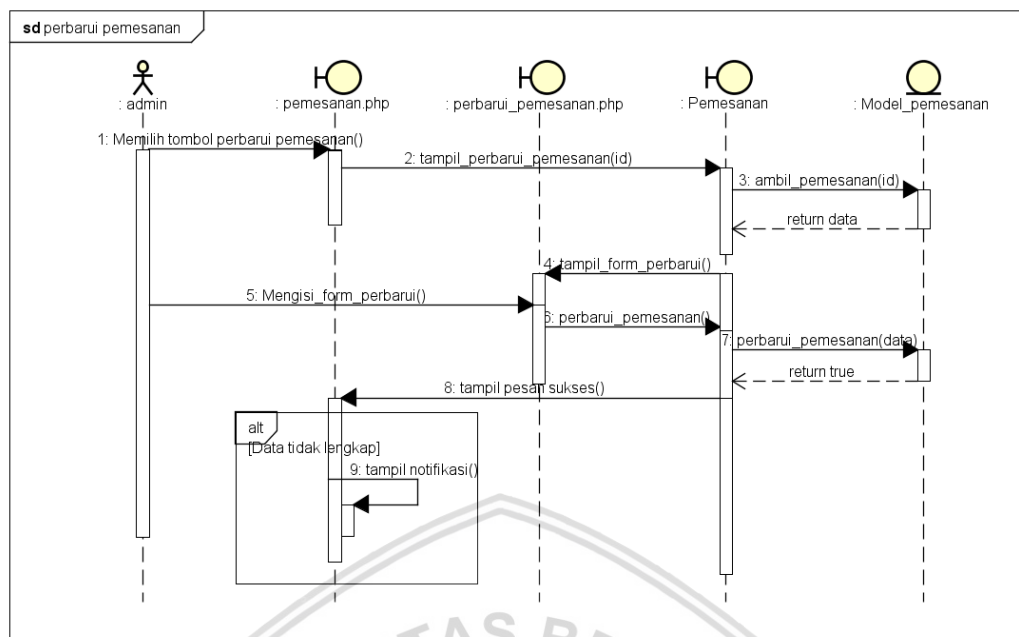
Sequence diagram tambah penjualan dimulai pada saat penjual dan kasir berada pada menu data penjualan, lalu sistem menampilkan daftar data penjualan dalam halaman dan sebuah fungsi untuk menambah data penjualan dalam halaman tambah data penjualan. Penjual memilih halaman tambah data penjualan, kemudian sistem akan memberi tampilan berupa *form* untuk menambah data penjualan yang harus memiliki isi. Kemudian dilanjutkan ke *controller* C_penjual dan M_penjualan. *Sequence diagram* tambah data penjualan ditunjukkan pada gambar 5.2.



Gambar 5.2 Sequence Diagram Tambah Penjualan

b. Perbarui data penjualan

Sequence diagram perbarui data penjualan dimulai pada saat penjual berada dalam menu data penjualan, lalu sistem akan menampilkan daftar data penjualan dalam halaman dan fungsi untuk memperbarui data penjualan pada fungsi perbarui data penjualan. Penjual atau kasir memilih fungsi perbarui data penjualan, kemudian sistem akan memberi tampilan berupa *form* untuk memperbarui data penjualan yang harus memiliki isi. Kemudian dilanjutkan ke *controller* C_penjual dan M_penjualan. *Sequence diagram* perbarui data penjualan ditunjukkan pada gambar 5.3.



Gambar 5.3 Sequence Diagram Perbarui Penjualan

3. *Sequence Diagram* mengelola data pegawai

a. Tambah data pegawai

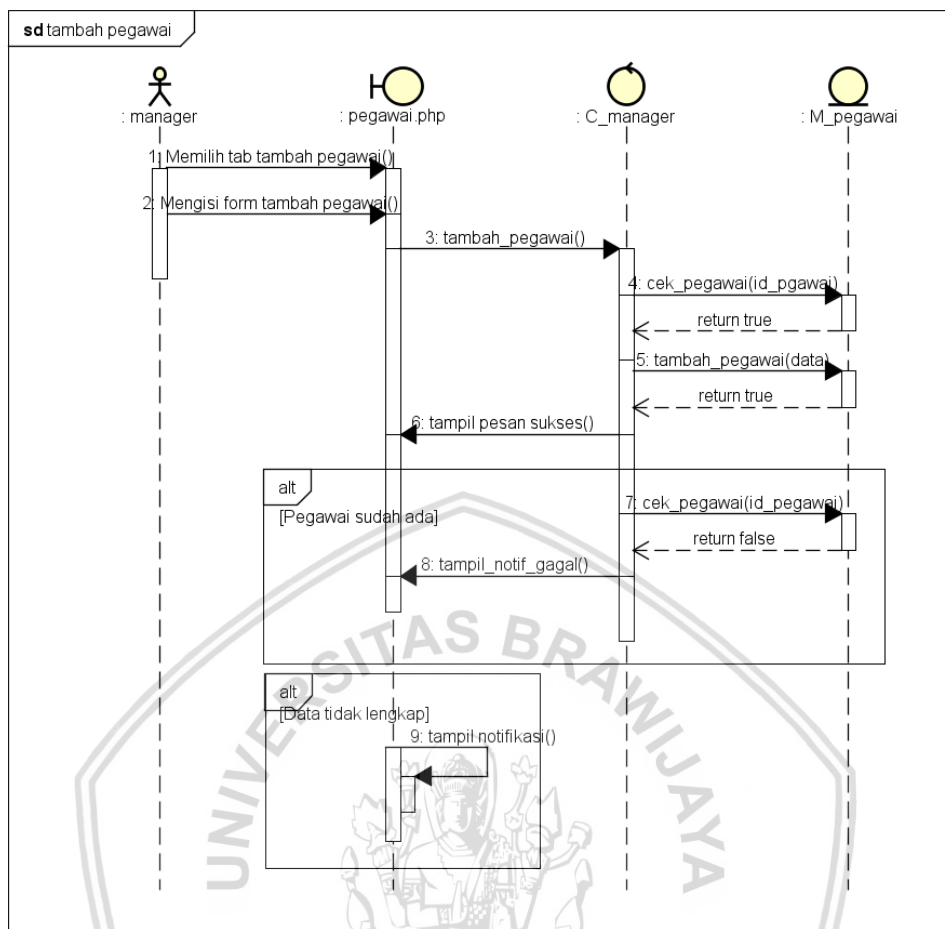
Sequence diagram tambah data pegawai dimulai pada saat *manager* berada dalam menu data pegawai, dan sistem akan menampilkan daftar data pegawai dalam halaman dan fungsi untuk menambah data pegawai dalam halaman tambah data pegawai. *Manager* memilih halaman tambah data pegawai, kemudian sistem akan memberi tampilan berupa *form* untuk tambah data pegawai yang harus memiliki isi. Kemudian dilanjutkan ke *controller* *C_manager* dan *M_pegawai*. *Sequence diagram* tambah data pegawai ditunjukkan pada gambar 5.4.

b. Hapus data pegawai

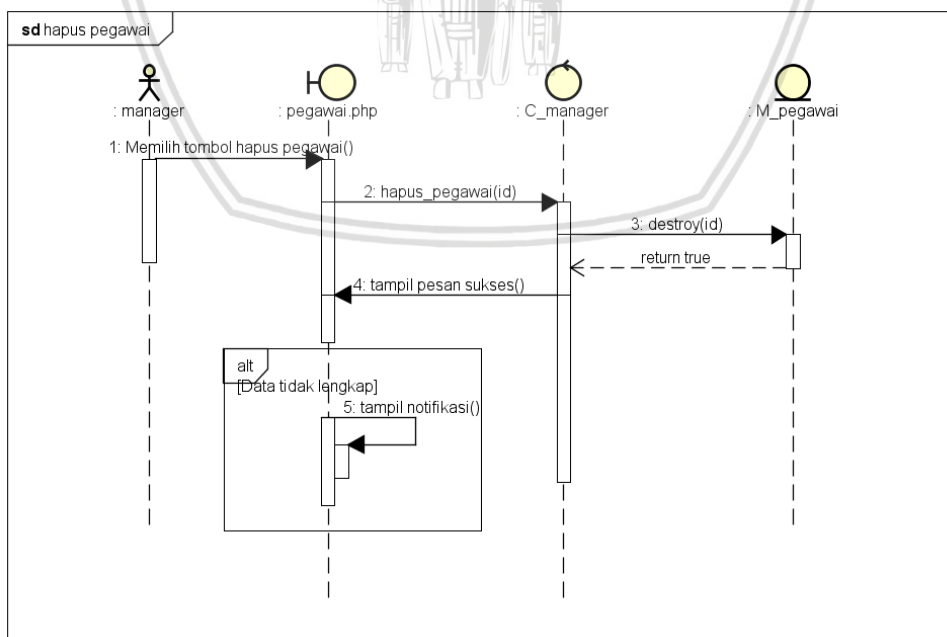
Pada *sequence diagram* hapus data pegawai dimulai pada saat *manager* berada dalam menu data pegawai dan *manager* melakukan penghapusan data pegawai dengan memilih fungsi hapus pegawai. *Sequence diagram* tambah data pegawai ditunjukkan pada gambar 5.5.

c. Perbarui data pegawai

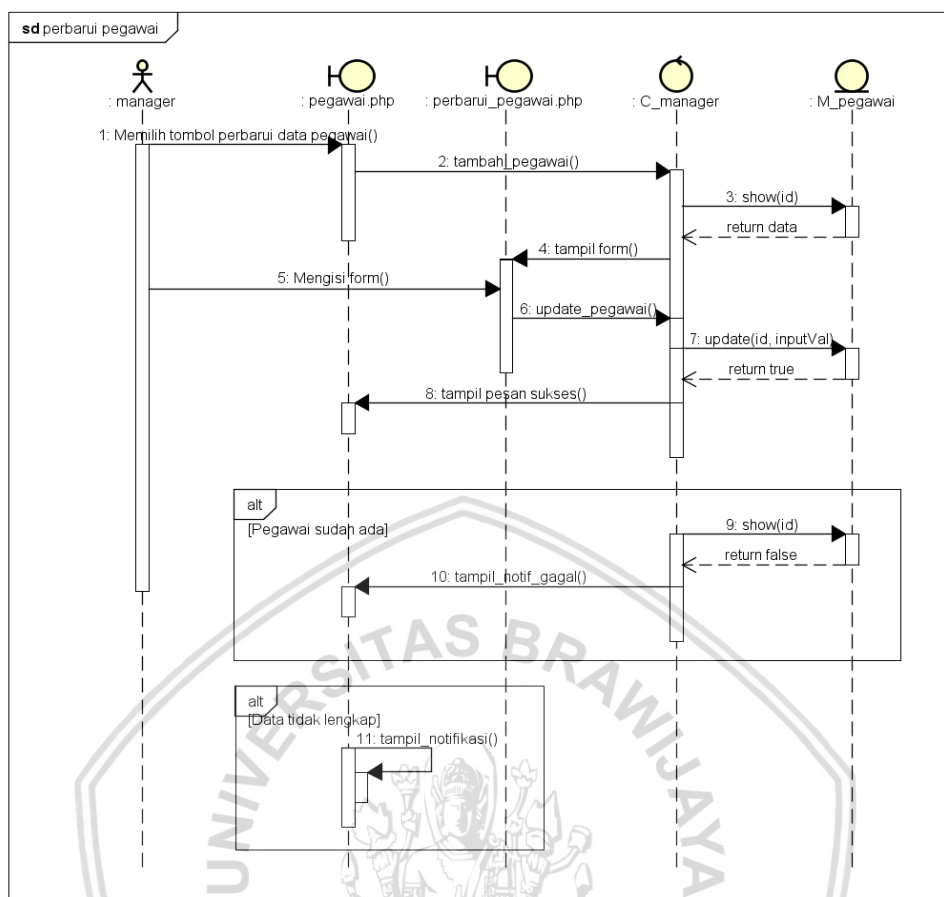
Sequence diagram perbarui data pegawai dimulai pada saat *manager* berada dalam menu data pegawai. Sistem akan menampilkan daftar data pegawai dalam halaman daftar data pegawai. *Manager* memilih fungsi perbarui dalam halaman daftar data pegawai. Lalu dilanjutkan ke dalam *controller* *C_manager* dan *M_pegawai*. *Sequence diagram* hapus data pegawai ditunjukkan pada gambar 5.6.



Gambar 5.4 Sequence Diagram Tambah Data Pegawai



Gambar 5.5 Sequence Diagram Hapus Data Pegawai



Gambar 5.6 Sequence Diagram Perbarui Pegawai

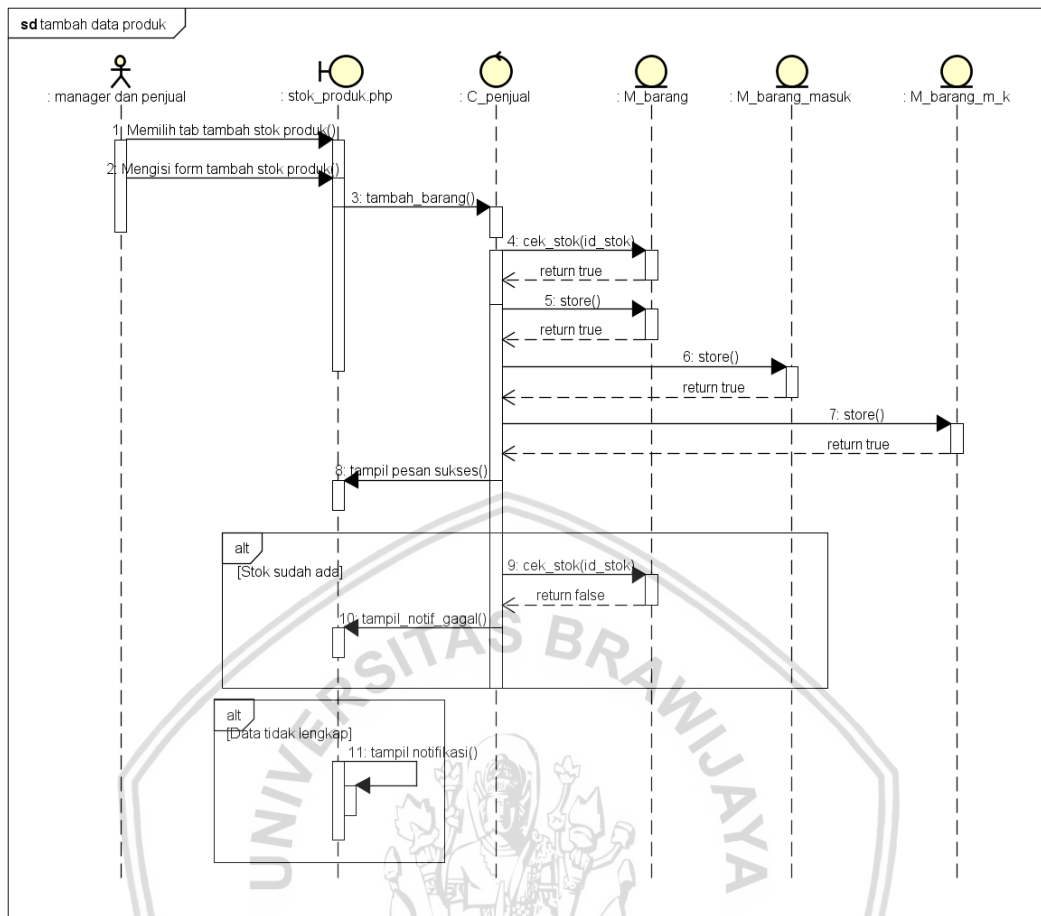
4. Sequence diagram mengelola data produk

a. Tambah data produk

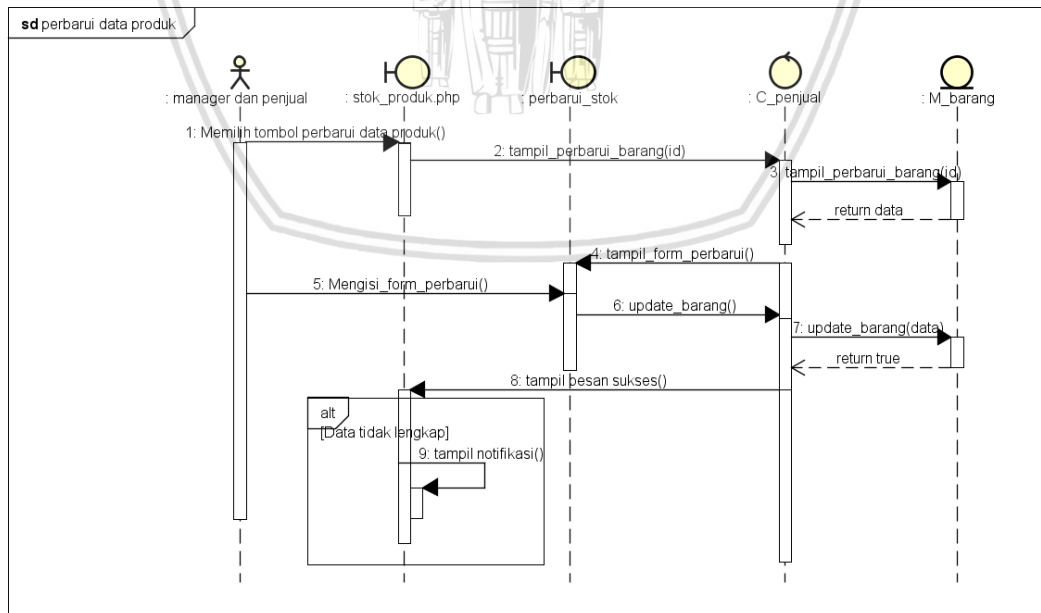
Sequence diagram tambah data produk dimulai pada saat manager dan penjual berada dalam menu data produk, dan sistem akan menampilkan daftar data produk dalam halaman dan fungsi untuk menambah data produk dalam halaman tambah data produk. *Manager* dan penjual memilih halaman tambah data produk, kemudian sistem akan memberi tampilan berupa *form* untuk menambah data produk yang harus memiliki isi. Kemudian dilanjutkan ke *controller* C_penjual dan M_barang. *Sequence diagram* tambah data produk ditunjukkan pada gambar 5.7.

b. Perbarui data produk

Sequence diagram perbarui data produk dimulai pada saat admin berada dalam menu data produk. Admin memilih fungsi perbarui data produk, kemudian sistem akan memberi tampilan berupa *form* untuk memperbarui data produk yang harus memiliki isi. Kemudian dilanjutkan ke *controller* C_penjual dan M_barang. *Sequence diagram* perbarui data produk ditunjukkan pada gambar 5.8.



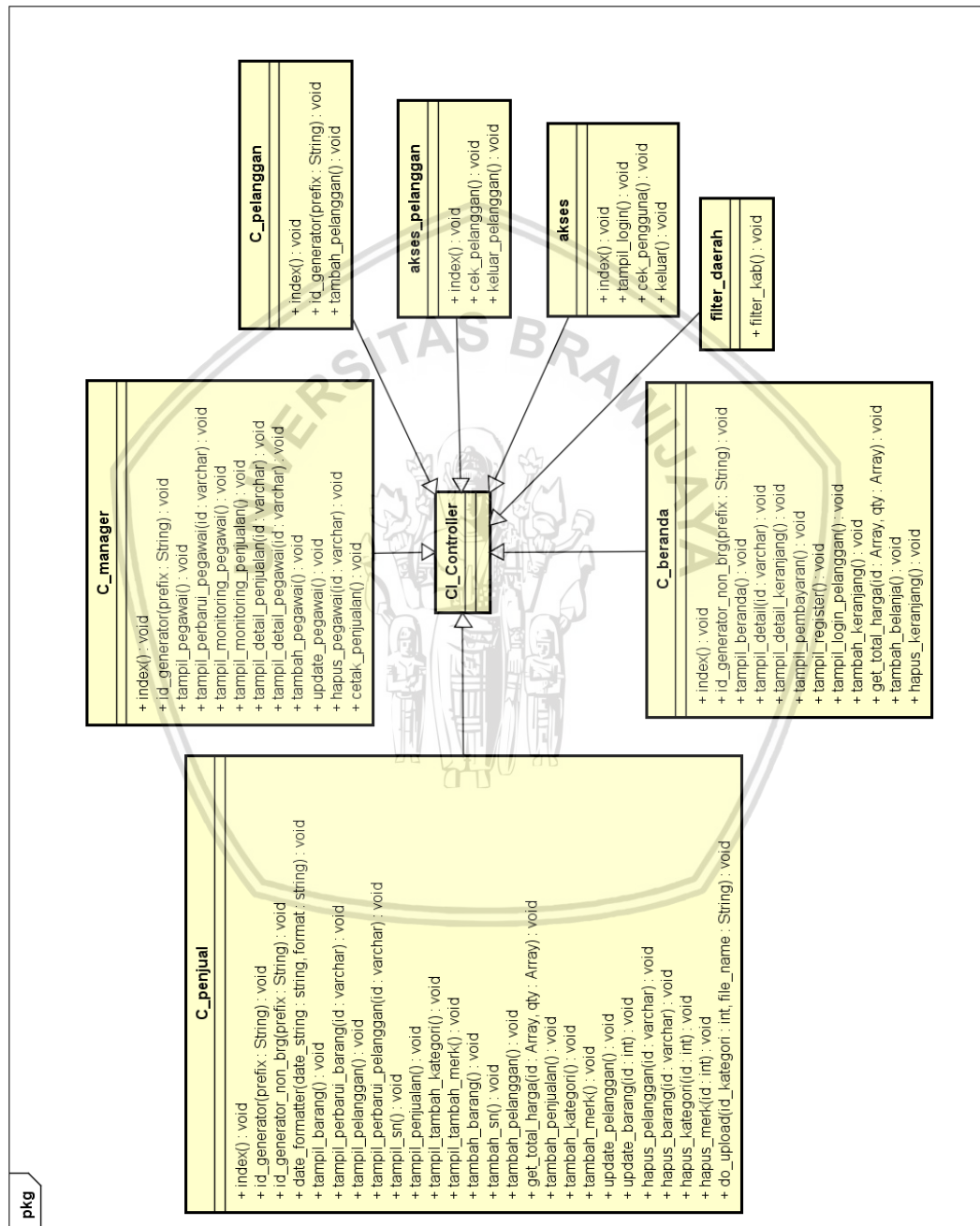
Gambar 5.7 Sequence Diagram Tambah Data Produk



Gambar 5.8 Sequence Diagram Perbarui Data Produk

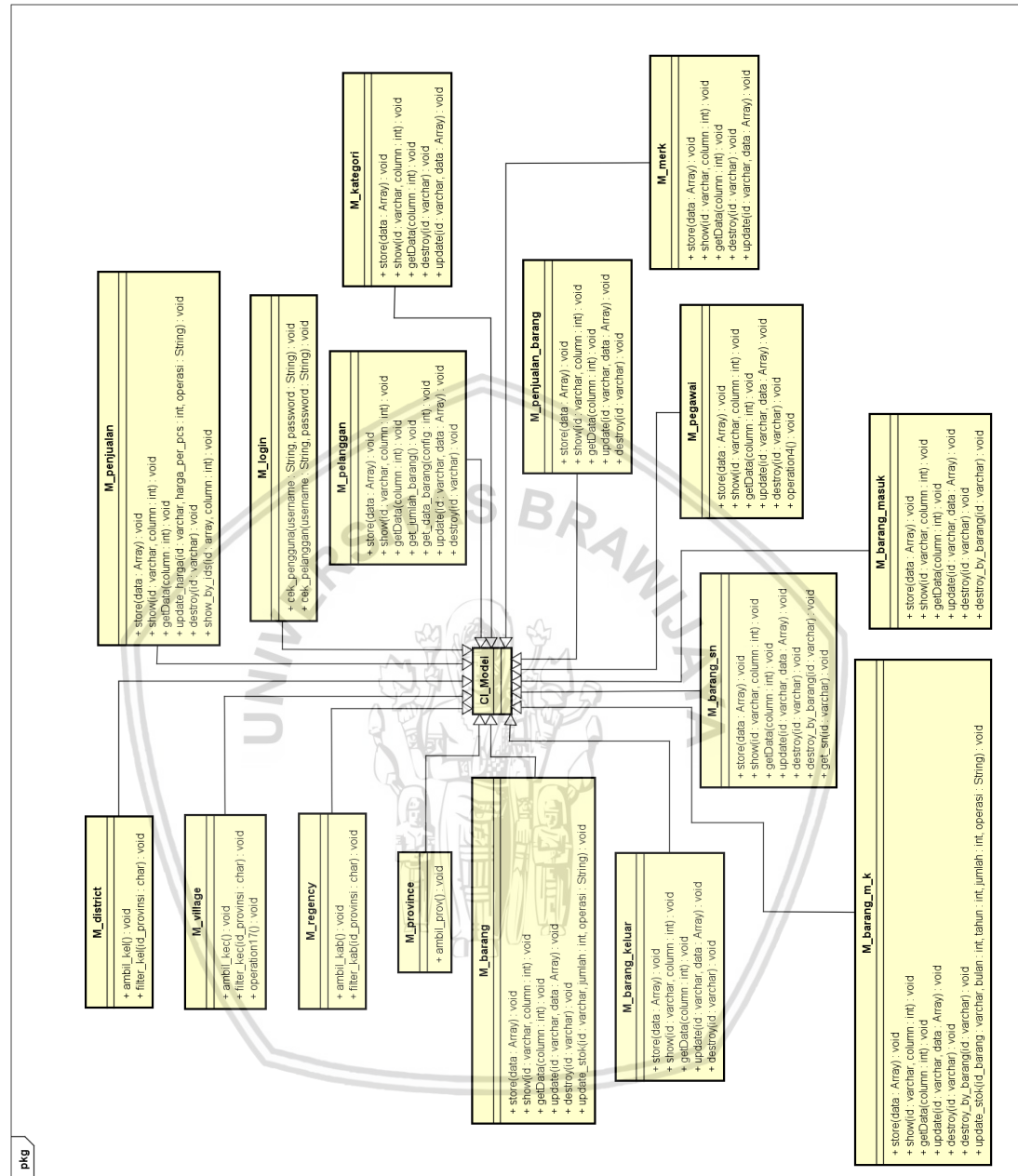
5.2 Perancangan Class Diagram

Perancangan *Class Diagram* merupakan gambaran struktur sistem dengan pendefinisian setiap kelas-kelas yang akan dibuat untuk membangun sistem. Perancangan *class diagram* dibuat dengan menyesuaikan dengan entitas-entitas yang sudah diidentifikasi pada sequence diagram dengan pola MVC pada kerangka kerja pemrograman PHP Codeigniter.



Gambar 5.9 Class Diagram Controller

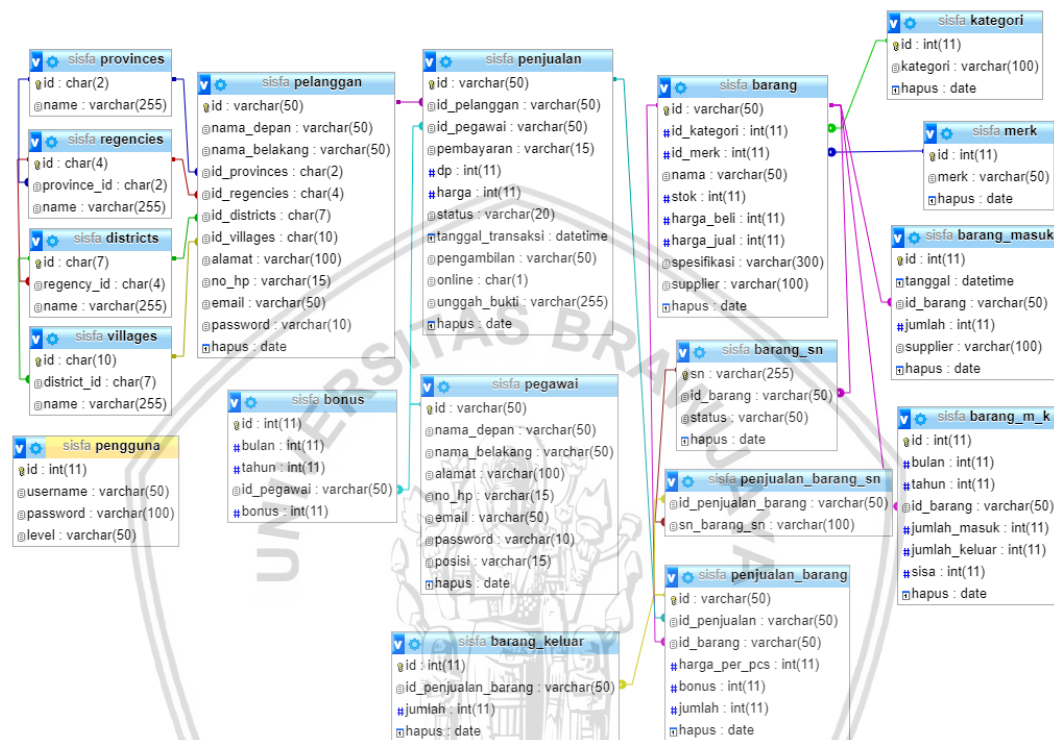
Berikut ini merupakan *model class diagram* yang ditunjukkan pada gambar 5.10.



Gambar 5.10 Class Diagram Model

5.3 Perancangan Basis Data

Perancangan basis data digunakan untuk membuat model atau rancangan yang menggunakan sejumlah tabel sebagaimana data direkam dalam sistem untuk menggambarkan hubungan proses antar data penjualan. Berdasarkan analisis *class diagram* yang telah dilakukan sebelumnya, maka didapatkan rancangan basis data yang direpresentasikan dalam bentuk *physical data model* seperti ditunjukkan pada gambar 5.11.



Gambar 5.11 Physical Data Model

5.4 Perancangan Komponen

Tahap perancangan komponen merupakan tahap untuk mendefinisikan struktur data dan menentukan algoritma yang digunakan dalam suatu komponen perangkat lunak. Dalam perancangan komponen hanya mencantumkan algoritma proses penjualan produk sehingga tidak semua algoritma dicantumkan.

5.4.1 Algoritma Method tambah_penjualan

Perancangan komponen fungsi tambah_penjualan() adalah algoritma dari salah satu fungsi sistem untuk melakukan penambahan data penjualan. Fungsi tambah_penjualan() merupakan bagian dari *controller* C_penjual. Tabel 5.1 dibawah merupakan algoritma dari fungsi tambah_penjualan().

Tabel 5.1 Algoritma Fungsi tambah_penjualan

1	Mulai
2	Menyimpan data penjualan ke dalam variabel
3	Menyimpan variabel-variabel kedalam array
4	Jika DP lebih dari 0
5	Deklarasi status menjadi DP
6	Lainnya
7	Deklarasi status menjadi lunas
8	Melakukan kontak dengan model untuk menyimpan variabel array penjualan ke dalam database sistem
9	Menampilkan halaman daftar data penjualan
10	Selesai

5.4.2 Algoritma Method update_pegawai

Perancangan komponen fungsi update_pegawai() adalah algoritma dari salah satu fungsi sistem untuk melakukan perbaruan data. Fungsi update_pegawai() merupakan bagian dari *controller* C_manager. Tabel 5.2 dibawah merupakan algoritma dari fungsi update_pegawai().

Tabel 5.2 Algoritma Fungsi update_pegawai

1	Mulai
2	Menyimpan data seorang pegawai ke dalam variabel
3	Jika nama pegawai sudah ada
4	Menampilkan pesan pegawai sudah ada
5	Lainnya
6	Melakukan kontak dengan model untuk menyimpan variabel data pegawai ke dalam database sistem
7	Menampilkan halaman daftar data pegawai
8	Selesai

5.5 Perancangan Antarmuka

Pada tahapan ini akan memberikan penjelasan mengenai perancangan antarmuka dari sistem informasi *sales force automation*.

5.5.1 Halaman Login

Halaman *login* merupakan halaman yang digunakan untuk akses masuk ke dalam sistem, pengguna harus memasukkan *username* dan *password* untuk dapat masuk ke dalam sistem. Pada gambar 5.12 merupakan hasil rancangan antarmuka login.

Gambar 5.12 Rancangan Halaman Login

5.5.2 Halaman Daftar Data Produk

Halaman daftar data produk merupakan halaman untuk menampilkan data produk keseluruhan yang sudah dimasukkan. Rancangan pada antarmuka halaman daftar data produk dapat digunakan untuk halaman daftar data lainnya, misalnya daftar data pelanggan, daftar data penjualan, dan lain-lain. Pada gambar 5.13 merupakan hasil rancangan antarmuka daftar data produk.

No	Kategori	Merk	Nama	Harga Beli	Harga Jual	Stok	Aksi
1	LAPTOP	ASUS	ROG	15000000	20000000	5	
2	PRINTER	HP	DESKJET	1000000	2000000	2	
3	PC	LENOVO	YOGA	10000000	15000000	1	

Gambar 5.13 Rancangan Halaman Daftar Data Produk

5.5.3 Halaman Tambah Data Produk

Halaman tambah data produk merupakan halaman untuk menambahkan data produk. Rancangan pada antarmuka halaman tambah data produk dapat digunakan untuk halaman tambah data lainnya, misalnya tambah data pelanggan, tambah data penjualan, dan lain-lain. Pada gambar 5.14 merupakan hasil rancangan antarmuka tambah data produk.

Logo		Pengguna			
Beranda Data Produk ▾ Data Produk & Tambah produk baru Tambah Kategori Tambah Merk Data Pelanggan Data Pegawai Penjualan Monitoring ▾ Laporan Penjualan Pencapaian Pegawai Laporan Barang keluar & masuk	Produk				
	<table border="1"> <tr> <td>Data Produk</td> <td>Tambah Produk</td> </tr> <tr> <td colspan="2"> Kategori <input type="text"/> Merk Produk <input type="text"/> Nama Produk <input type="text"/> Harga Beli <input type="text"/> Harga Jual <input type="text"/> Jumlah Produk <input type="text"/> Gambar <input type="text"/> <input type="button" value="Simpan"/> <input type="button" value="Ulang"/> </td> </tr> </table>		Data Produk	Tambah Produk	Kategori <input type="text"/> Merk Produk <input type="text"/> Nama Produk <input type="text"/> Harga Beli <input type="text"/> Harga Jual <input type="text"/> Jumlah Produk <input type="text"/> Gambar <input type="text"/> <input type="button" value="Simpan"/> <input type="button" value="Ulang"/>
Data Produk	Tambah Produk				
Kategori <input type="text"/> Merk Produk <input type="text"/> Nama Produk <input type="text"/> Harga Beli <input type="text"/> Harga Jual <input type="text"/> Jumlah Produk <input type="text"/> Gambar <input type="text"/> <input type="button" value="Simpan"/> <input type="button" value="Ulang"/>					

Gambar 5.14 Rancangan Halaman Tambah Data Produk

5.5.4 Halaman Beranda Pelanggan

Halaman beranda pelanggan merupakan halaman untuk menampilkan data produk bagi pelanggan. Pada gambar 5.15 merupakan hasil rancangan antarmuka beranda pelanggan.

Header		
Slider		
Foto produk Detail Produk	Foto produk Detail Produk	Foto produk Detail Produk
Foto produk Detail Produk	Foto produk Detail Produk « 1 2 3 4 5 »	Foto produk Detail Produk
Footer		

Gambar 5.15 Rancangan Halaman Beranda Pelanggan

BAB 1 IMPLEMENTASI

6.1 Spesifikasi Sistem

Spesifikasi sistem merupakan lingkungan implementasi yang digunakan dalam pembuatan sistem ini yang terdiri dari spesifikasi perangkat keras dan perangkat lunak.

6.1.1 Spesifikasi Perangkat Keras

Penelitian ini penulis menggunakan sebuah laptop karena sistem yang dibangun tidak berkomunikasi dengan perangkat keras yang lain. Sistem *sales force automation* merupakan sistem informasi yang berbasis *website*, sehingga memerlukan koneksi internet untuk mengakses. Spesifikasi perangkat keras yang digunakan dalam pembuatan sistem *Sales Force Automation* pada Brandeet IT Store Malang dapat dilihat pada Tabel

Tabel 6.1 Spesifikasi Perangkat Keras

Unit Komputasi	HP Pavillion x360 13-u033TU
CPU	Core i5-6200 2.30GHz
Kapasitas RAM	8 GB
Kapasitas Penyimpanan	256 GB
Kartu Grafis	Intel HD Graphics 520
Resolusi Layar	1366 x 768

6.1.2 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam pembuatan sistem *Sales Force Automation* pada Brandeet IT Store Malang dapat dilihat pada Tabel

Tabel 6.2 Spesifikasi Perangkat Lunak

Sistem Operasi	Windows 10 62-bit
Bahasa Pemrograman	PHP
Framework	CodeIgniter
Editor Kode Program	Sublime Text 3
Web Service	Xampp v3.2.2

6.2 Implementasi Algoritma

Di dalam implementasi algoritma ini hanya mencantumkan proses pengelolaan data yang telah dicantumkan algoritmanya pada sub-bab perancangan komponen sehingga tidak semua fungsi dicantumkan.

6.2.1 Algoritma Mengelola Penjualan

Fungsi pada *controller* penjual yang berfungsi untuk menambahkan data penjualan adalah fungsi `tambah_penjualan()`. Data penjualan akan dimasukkan ke dalam database. Pada tabel dibawah merupakan implementasi algoritma dari fungsi `tambah_penjualan` dalam *controller* `C_penjual`.

Tabel 6.3 Fungsi tambah_penjualan

```

1 public function tambah_penjualan() {
2     $inputVal = $this->input->post();
3     $inputVal['id'] = $this->id_generator_non_brg('JUAL');
4     $total_harga=$this->get_total_harga($inputVal['id_barang'],
5     $inputVal['jumlah']);
6
7     $val_penjualan = array();
8     $val_penjualan_barang = array();
9     $val_barang_keluar = array();
10
11     $val_penjualan['id'] = $inputVal['id'];
12     $val_penjualan['id_pelanggan'] = $inputVal['id_pelanggan'];
13     $val_penjualan['pembayaran'] = $inputVal['pembayaran'];
14     $val_penjualan['dp'] = $inputVal['dp'];
15     $val_penjualan['harga'] = $total_harga;
16     if ($inputVal['dp'] > 0) {
17         $val_penjualan['status'] = 'dp';
18     } else {
19         $val_penjualan['status'] = 'lunas';
20     }
21     $val_penjualan['tanggal_transaksi'] = date('Y-m-d H:i:s');
22     $val_penjualan['pengambilan'] = $inputVal['pengambilan'];
23     $val_penjualan['online'] = '0';
24
25     $this->db->trans_begin();
26     $this->M_penjualan->store($val_penjualan);
27     foreach ($inputVal['id_barang'] as $key => $value) {
28         $jual = $this->M_barang->show($value, 'harga_jual');
29         $val_penjualan_barang['id'] = $this-
30         >id_generator_non_brg('JUALBRG');
31         $sar_harga = $jual->result_array();
32         $harga = $sar_harga[0]['harga_jual'];
33         $val_penjualan_barang['id_penjualan'] = $inputVal['id'];
34         $val_penjualan_barang['id_barang'] = $value;
35         $val_penjualan_barang['harga_per_pcs'] = $harga;
36         $val_penjualan_barang['bonus'] = 1;

```


Tabel 6.3 Fungsi tambah_penjualan (lanjutan)

```

35     $val_penjualan_barang['jumlah'] = $inputVal['jumlah'][$key];
36
37     $val_barang_keluar['id_penjualan_barang'] =
$val_penjualan_barang['id'];
38     $val_barang_keluar['jumlah'] =
$val_penjualan_barang['jumlah'];
39
    $val_Barang_m_k['bulan'] = $this-
>date_formatter($val_penjualan['tanggal_transaksi'], 'm');
40     $val_Barang_m_k['tahun'] = $this-
>date_formatter($val_penjualan['tanggal_transaksi'], 'Y');
41     $val_Barang_m_k['id_barang'] = $value;
42     $val_Barang_m_k['jumlah_keluar'] =
$val_barang_keluar['jumlah'];
43
44     $this->M_penjualan_barang->store($val_penjualan_barang);
45     $this->M_barang_keluar->store($val_barang_keluar);
46     $this->M_barang_m_k->update_stok($value,
$val_Barang_m_k['bulan'], $val_Barang_m_k['tahun'],
$val_Barang_m_k['jumlah_keluar'], 'keluar');
47     $this->M_barang->update_stok($value,
$val_Barang_m_k['jumlah_keluar'], 'keluar');
    }
48
49     if ($this->db->trans_status() === FALSE) {
50         $this->db->trans_rollback();
51         $url = base_url('penjual/C_penjual/tampil_penjualan');
52         header("Location: $url");
53     } else {
54         $this->db->trans_commit();
55         $url = base_url('penjual/C_penjual/tampil_penjualan');
56         header("Location: $url");
57     }
58 }

```

6.2.2 Algoritma Mengelola Pegawai

Fungsi `perbarui_pegawai()` berisi *query* untuk memperbarui data pegawai, fungsi ini ditunjukkan dalam tabel dibawah ini.

Tabel 6.4 Fungsi update_pegawai

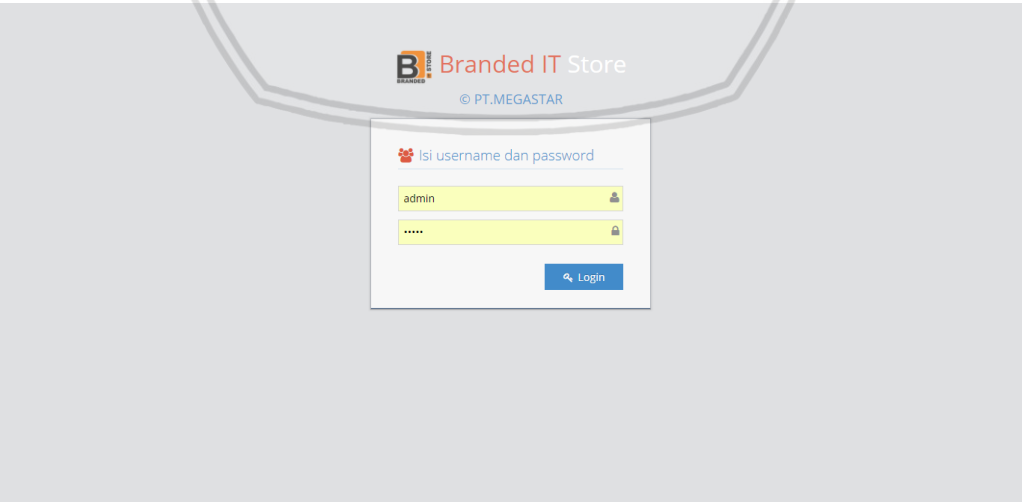
1	public function update_pegawai() {
2	\$inputVal = \$this->input->post();
3	\$id = \$this->input->post('id');
4	unset(\$inputVal['id']);
5	
6	\$this->db->trans_begin();
7	\$this->M_pegawai->update(\$id, \$inputVal);
8	if (\$this->db->trans_status() === FALSE) {
9	\$this->db->trans_rollback();
10	\$url = base_url() . 'manager/C_manager/tampil_pegawai';
11	header("Location: \$url");
12	} else {
13	\$this->db->trans_commit();
14	\$url = base_url() . 'manager/C_manager/tampil_pegawai';
15	header("Location: \$url");
16	}
17	}

6.3 Implementasi Antarmuka

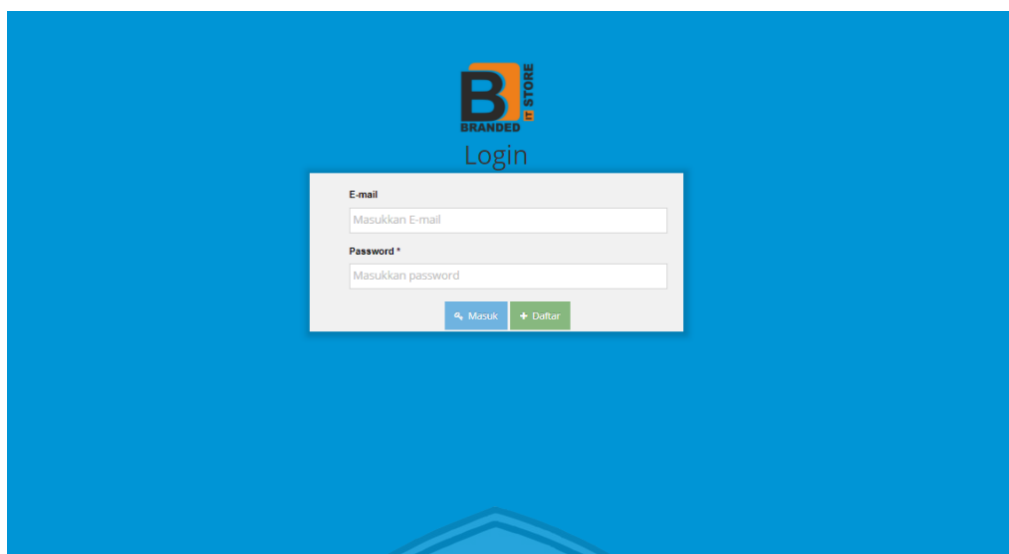
Berikut ini merupakan beberapa potongan gambar hasil implementasi antarmuka dari sistem informasi *sales force automation*.

6.3.1 Halaman Login

Halaman *login* menampilkan halaman untuk pengguna agar dapat masuk kedalam sistem. Pada gambar 6.1 menunjukan halaman login untuk admin dan gambar 6.2 halaman login untuk pelanggan.



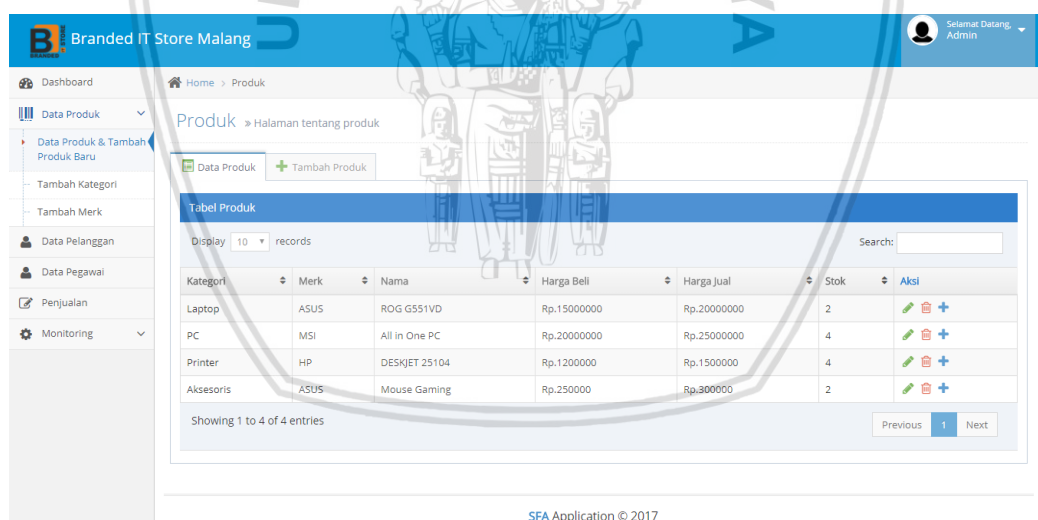
Gambar 6.1 Hasil Implementasi Halaman Login Admin



Gambar 6.2 Hasil Implementasi Halaman Login Pelanggan

6.3.2 Halaman Kelola Data Produk

Halaman kelola data produk menampilkan halaman yang dapat diakses oleh *manager* dan penjual yang berguna untuk mengelola data-data produk yang ada. Fungsi kelola data produk dalam sistem ini meliputi menambah dan memperbarui. Pada gambar 6.3 menunjukan halaman kelola data produk.



Gambar 6.3 Hasil Implementasi Halaman Kelola Data Produk

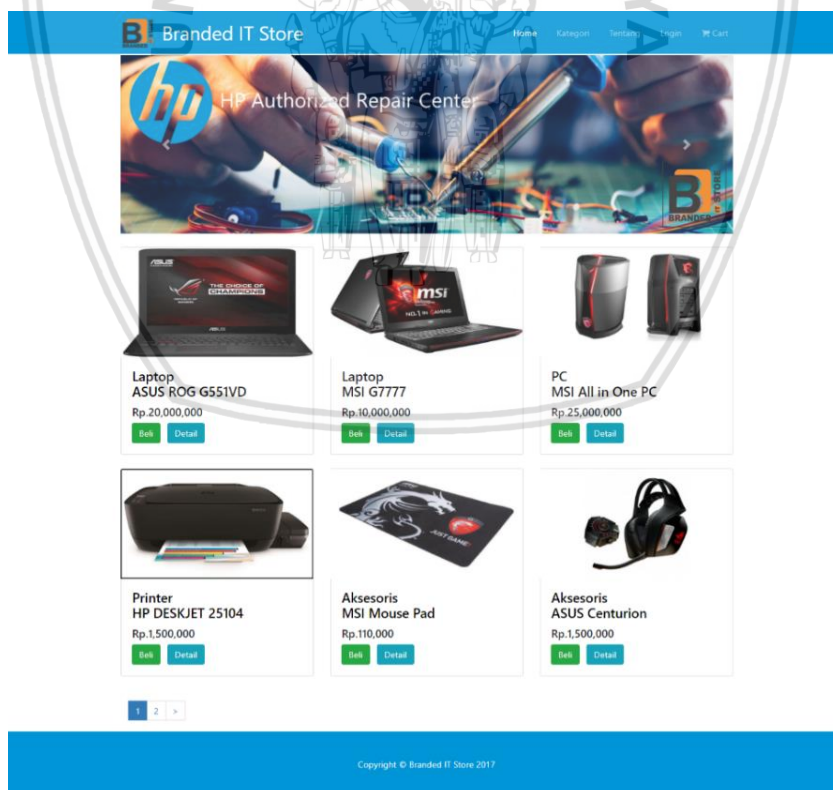
6.3.3 Halaman Kelola Data Penjualan

Halaman kelola data penjualan merupakan halaman yang dapat diakses oleh penjual dan kasir yang berguna untuk mengelola data-data penjualan yang masuk. Fungsi kelola data penjualan dalam sistem ini meliputi menambah dan memperbarui data penjualan. Pada gambar 6.4 menunjukan halaman kelola data penjualan.

Gambar 6.4 Hasil Implementasi Halaman Kelola Data Penjualan

6.3.4 Halaman Beranda Pelanggan

Halaman beranda pelanggan merupakan halaman yang dapat diakses oleh pelanggan yang berguna untuk melihat produk-produk yang dimiliki toko bagi pelanggan. Pada halaman beranda pelanggan dapat melakukan pembelian dan melihat detail produk yang ada. Pada gambar 6.5 menunjukkan halaman beranda pelanggan.



Gambar 6.5 Hasil Implementasi Halaman Beranda Pelanggan

6.3.5 Halaman Rincian Belanja

Halaman rincian belanja merupakan halaman yang dapat diakses oleh pelanggan yang berguna untuk melihat rincian produk yang sudah pelanggan pilih untuk dibeli dan mengisi form yang diperlukan. Pada gambar 6.6 menunjukkan halaman rincian belanja.

	Harga	Jumlah	Total
Laptop ASUS ROG G551VD i7-7700HQ, 8GB RAM, 1TB HDD, NVIDIA GEFORCE GTX1080	Rp.20000000	1	Rp. 20000000

Subtotal Rp. 20000000

Pengambilan:

Pembayaran:

Unggah bukti transfer: No file chosen

Copyright © Branded IT Store 2017

Gambar 6.6 Hasil Implementasi Halaman Rincian Belanja

6.4 Evaluasi Pelanggan

Setelah implementasi, *prototyping* sistem informasi SFA disampaikan kepada pelanggan untuk melakukan penyesuaian *prototyping* dengan spesifikasi kebutuhan yang sudah disepakati diawal. Proses evaluasi pelanggan dilakukan dengan cara pengembang mendemonstrasikan sistem informasi yang sudah dibuat. Setelah melakukan demonstrasi sistem informasi dilakukan wawancara untuk mengetahui kesesuaian spesifikasi kebutuhan dengan *prototyping* yang sudah didemonstrasikan sebelumnya. Berdasarkan hasil wawancara yang terlampir pada lampiran A hasil wawancara didapatkan beberapa penambahan fungsi dari fitur yang sudah disepakati diawal. Penambahan fungsi berada pada fitur melihat pencapaian pegawai dan melihat laporan penjualan yang terlampir pada lampiran b hasil perulangan *prototyping*.

BAB 7 PENGUJIAN

Bab ini akan menjelaskan proses pengujian. Tujuan dari bab ini untuk menemukan kesalahan yang bisa terjadi.

7.1 Validation Testing

Metode *validation testing* merupakan bagian dari *black-box testing*. *Validation testing* dibutuhkan mengetahui sistem yang dibuat sudah berjalan sesuai dengan spesifikasi kebutuhan yang ada.

Tabel 7.1 Menambah Data Produk

Nama Kasus Uji	Menambah data produk.
Objek Uji	F01-SFA-F-01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan sistem dapat menambah data produk.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu data produk. 2. Aktor memilih <i>tab</i> tambah produk. 3. Aktor mengisi form data produk. 4. Aktor memilih tombol simpan.
Hasil Yang Diharapkan	Sistem dapat menyimpan data produk ke dalam database dan menampilkan notifikasi data berhasil ditambah.
Hasil Pengujian	Sistem menyimpan data produk ke dalam sistem dan menampilkan notifikasi data berhasil ditambah.
Status Validasi	Valid.

Tabel 7.2 Menambah Data Produk Alternatif 1: Data Produk Yang Dimasukkan Belum Lengkap

Nama Kasus Uji	Menambah data produk.
Objek Uji	F01-SFA-F-01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan sistem dapat menampilkan notifikasi jika kolom dalam form tambah data produk belum lengkap.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu data produk. 2. Aktor memilih <i>tab</i> tambah produk. 3. Aktor mengisi form data produk. 4. Aktor memilih tombol simpan.
Hasil Yang Diharapkan	Sistem dapat menampilkan notifikasi bahwa ada kolom form yang kosong harus diisi.
Hasil Pengujian	Sistem menampilkan notifikasi bahwa ada kolom form yang kosong harus diisi.
Status Validasi	Valid.

Tabel 7.3 Memperbarui Data Produk

Nama Kasus Uji	Memperbarui data produk.
Objek Uji	F01-SFA-F-02
Tujuan Pengujian	Pengujian dilakukan untuk memastikan sistem dapat memperbarui data produk.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu data produk. 2. Aktor memilih tombol perbarui Produk. 3. Aktor merubah isi form data produk. 4. Aktor memilih tombol simpan
Hasil Yang Diharapkan	Sistem dapat memperbarui data produk yang sudah diubah dan menampilkan notifikasi data berhasil diperbarui.
Hasil Pengujian	Sistem berhasil memperbarui data produk dan menampilkan notifikasi data berhasil diperbarui.
Status Validasi	Valid.

Tabel 7.4 Menambah Data Penjualan

Nama Kasus Uji	Menambah data penjualan.
Objek Uji	F02-SFA-F-01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan sistem dapat menambah data penjualan.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu penjualan. 2. Aktor berada pada <i>tab</i> tambah penjualan. 3. Aktor mengisi form data penjualan. 4. Aktor memilih tombol simpan.
Hasil Yang Diharapkan	Sistem dapat menyimpan data produk ke dalam database dan menampilkan notifikasi data berhasil ditambah.
Hasil Pengujian	Sistem menyimpan data penjualan ke dalam database dan menampilkan notifikasi data berhasil ditambah.
Status Validasi	Valid.

Tabel 7.5 Memperbarui Data Penjualan

Nama Kasus Uji	Memperbarui data penjualan.
Objek Uji	F02-SFA-F-02
Tujuan Pengujian	Pengujian dilakukan untuk memastikan sistem dapat memperbarui data penjualan.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu penjualan. 2. Aktor memilih <i>tab</i> data penjualan. 3. Aktor memilih tombol perbarui data penjualan. 4. Aktor merubah isi form data penjualan 5. Aktor memilih tombol simpan.

Tabel 7.5 Memperbarui Data Penjualan(lanjutan)

Hasil Yang Diharapkan	Sistem dapat memperbarui data penjualan ke dalam database dan menampilkan notifikasi data berhasil diperbarui.
Hasil Pengujian	Sistem memperbarui data penjualan dan menampilkan notifikasi data berhasil diperbarui.
Status Validasi	Valid.

Tabel 7.6 Menampilkan Data Penjualan

Nama Kasus Uji	Menampilkan data penjualan.
Objek Uji	F02-SFA-F-03
Tujuan Pengujian	Pengujian dilakukan untuk memastikan sistem dapat menampilkan data penjualan.
Prosedur Uji	1. Aktor berada pada menu penjualan. 2. Aktor memilih <i>tab</i> data penjualan.
Hasil Yang Diharapkan	Sistem dapat menampilkan data penjualan.
Hasil Pengujian	Sistem berhasil menampilkan data penjualan.
Status Validasi	Valid.

Tabel 7.7 Menambah Data Pelanggan

Nama Kasus Uji	Menambah data pelanggan.
Objek Uji	F03-SFA-F-01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan sistem dapat menambah data pelanggan.
Prosedur Uji	1. Aktor memilih menu data pelanggan. 2. Aktor memilih <i>tab</i> tambah data pelanggan. 3. Aktor mengisi form data pelanggan. 4. Aktor memilih tombol simpan.
Hasil Yang Diharapkan	Sistem dapat menyimpan data pelanggan ke dalam database dan menampilkan notifikasi data berhasil ditambah.
Hasil Pengujian	Sistem menyimpan data pelanggan ke dalam database dan menampilkan notifikasi data berhasil ditambah.
Status Validasi	Valid.

Tabel 7.8 Menambah Data Pelanggan Alternatif 1: Data Pelanggan Yang Dimasukkan Sudah Ada

Nama Kasus Uji	Menambah data Pelanggan.
Objek Uji	F03-SFA-F-01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan sistem dapat menampilkan notifikasi jika data pelanggan sama.

Tabel 7.8 Menambah Data Pelanggan Alternatif 1: Data Pelanggan Yang Dimasukkan Sudah Ada(lanjutan)

Prosedur Uji	1. Aktor memilih menu data pelanggan. 2. Aktor memilih <i>tab</i> tambah data pelanggan. 3. Aktor mengisi form data pelanggan. 4. Aktor memilih tombol simpan.
Hasil Yang Diharapkan	Sistem dapat menampilkan notifikasi bahwa data pelanggan sudah ada.
Hasil Pengujian	Sistem menampilkan notifikasi bahwa data pelanggan sudah ada.
Status Validasi	Valid.

Tabel 7.9 Memperbarui Data Pelanggan

Nama Kasus Uji	Memperbarui data pelanggan.
Objek Uji	F03-SFA-F-02
Tujuan Pengujian	Pengujian dilakukan untuk memastikan sistem dapat memperbarui data pelanggan.
Prosedur Uji	1. Aktor memilih menu pelanggan. 2. Aktor memilih <i>tab</i> data pelanggan. 3. Aktor memilih tombol perbarui data pelanggan. 4. Aktor merubah isi form data pelanggan. 5. Aktor memilih tombol simpan.
Hasil Yang Diharapkan	Sistem dapat memperbarui data pelanggan ke dalam database dan menampilkan notifikasi data berhasil diperbarui.
Hasil Pengujian	Sistem memperbarui data pelanggan dan menampilkan notifikasi data berhasil diperbarui.
Status Validasi	Valid.

Tabel 7.10 Menambah Data Pegawai

Nama Kasus Uji	Menambah data pegawai.
Objek Uji	F04-SFA-F-01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan sistem dapat menambah data pegawai.
Prosedur Uji	1. Aktor memilih menu data pegawai. 2. Aktor memilih <i>tab</i> tambah data pegawai. 3. Aktor mengisi form data pegawai. 4. Aktor memilih tombol simpan.
Hasil Yang Diharapkan	Sistem dapat menyimpan data pegawai ke dalam database dan menampilkan notifikasi data ditambah.
Hasil Pengujian	Sistem menyimpan data pegawai ke dalam database dan menampilkan notifikasi data berhasil ditambah.
Status Validasi	Valid.

Tabel 7.11 Menghapus Data Pegawai

Nama Kasus Uji	Menghapus data pegawai.
Objek Uji	F04-SFA-F-03
Tujuan Pengujian	Pengujian dilakukan untuk memastikan sistem dapat menambah data pegawai.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor memilih menu data pegawai. 2. Aktor memilih tombol hapus data pegawai. 3. Aktor mengkonfirmasi penghapusan data pegawai.
Hasil Yang Diharapkan	Sistem dapat menghapus data pegawai ke dalam database dan menampilkan pesan notifikasi data berhasil ditambah.
Hasil Pengujian	Sistem berhasil menghapus data pegawai dan menampilkan notifikasi data berhasil dihapus.
Status Validasi	Valid.

Tabel 7.12 Melihat Barang

Nama Kasus Uji	Melihat barang.
Objek Uji	F09-SFA-F-01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan pelanggan dapat melihat detail barang.
Prosedur Uji	<ol style="list-style-type: none"> 3. Aktor berada pada menu beranda pelanggan. 4. Aktor memilih tombol detail pada salah satu barang.
Hasil Yang Diharapkan	Sistem dapat menampilkan halaman detail barang yang dipilih pelanggan.
Hasil Pengujian	Sistem berhasil menampilkan halaman detail barang.
Status Validasi	Valid.

Tabel 7.13 Membeli Barang

Nama Kasus Uji	Membeli barang.
Objek Uji	F10-SFA-F-01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan pelanggan dapat melakukan pembelian barang.
Prosedur Uji	<ol style="list-style-type: none"> 1. Aktor berada pada menu beranda pelanggan. 2. Aktor memilih tombol beli pada salah satu barang. 3. Aktor memilih menu cart. 4. Aktor memilih tombol submit. 5. Aktor mengisi form rincian. 6. Aktor memilih tombol simpan.
Hasil Yang Diharapkan	Sistem dapat menyimpan data pembelian pelanggan ke dalam database dan menampilkan menu beranda pelanggan.

Tabel 7.13 Membeli Barang (lanjutan)

Hasil Pengujian	Sistem menyimpan data pembelian pelanggan ke dalam database dan menampilkan menu beranda pelanggan.
Status Validasi	Valid.

7.2 User Acceptance Testing

User Acceptance Testing dilakukan oleh pelanggan untuk memastikan bahwa sistem sudah memenuhi kriteria spesifikasi kebutuhan yang sudah dijelaskan sebelumnya dan dapat diterima oleh pengguna. UAT akan dilakukan kepada stakeholder ataupun yang bertanggung jawab terhadap sistem.

Tabel 7.14 Stakeholder yang terlibat pada *User Acceptance Testing*

No	Stakeholders	Alasan
1	Manager	Supaya dapat mengelola data pelanggan, pegawai dan penjualan
2	Penjual	Supaya dapat mengoperasikan sistem yang akan digunakan
3	Kasir	Supaya dapat mengoperasikan sistem yang akan digunakan
4	Pelanggan	Supaya memudahkan untuk mendapatkan informasi

Kriteria user acceptance testing yang digunakan menyesuaikan dengan kebutuhan spesifikasi pelanggan yaitu, *functional correctness and completeness*, *confidentiality and availability*, dan *usability*. Pengujian akan dilaksanakan dengan menghampiri *tester*(penguji) untuk melakukan pengujian terhadap sistem yang telah dibuat (Naik & Kshirasagar, 2008).

Tabel 7.15 Kriteria UAT

No	Kriteria	Definisi
1	<i>Functional Correctness and Completeness</i>	Pengujian untuk mengetahui apakah sistem yang dibuat sudah sesuai dengan kebutuhan yang diinginkan pengguna dan semua fitur yang dijelaskan dalam spesifikasi persyaratan harus ada dalam sistem yang dikirimkan
2	<i>Usability</i>	Evaluasi produk atau layanan dengan mengujinya untuk mengetahui bahwa sistem dapat dengan mudah digunakan dan kepuasan pengguna.

Tabel 7.15 Kriteria UAT(lanjutan)

No	Kriteria	Definisi
3	<i>Confidentiality and Availability</i>	<p>Confidentiality adalah sebuah aturan yang memiliki kemampuan untuk membatasi hak akses terhadap informasi tertentu.</p> <p>Availability adalah jaminan akses terhadap data yang dapat dipercaya berkaitan dengan informasi yang dipegang oleh orang-orang yang berwenang</p>

Sumber: Naik & Kshirasagar [2008]

Selanjutnya masing-masing tester akan diberikan pertanyaan sesuai dengan fungsi yang diberikan kepada masing-masing tester. Tabel 7.10 menunjukkan daftar pertanyaan yang akan ditanyakan kepada kepada tester beserta kriteria pertanyaan tersebut. Sebelum tester diberikan pertanyaan, pengembang akan mendemonstrasikan program menyesuaikan dengan task yang akan diberikan ditunjukkan pada tabel 7.10.

Tabel 7.16 Task yang Perlu Dikerjakan Tester

No	Tugas yang dikerjakan	Tester
1	Berada pada halaman beranda	Pelanggan, Manager, Penjual, dan Kasir
2	Melihat produk pada halaman beranda	
3	Memilih dan melihat detail produk	
4	Melakukan login	
5	Melakukan pembelian produk	
6	Melihat laporan penjualan	Manager
7	Melihat pencapaian pegawai	
8	Mengelola data pegawai	
9	Mengelola data pelanggan	Manager dan Penjual
10	Mengelola data penjualan	Penjual dan Kasir
11	Mengelola data produk	Manager, Penjual, dan Kasir

Pertanyaan akan diberikan kepada masing-masing tester, yaitu pertanyaan untuk manager, untuk penjual, untuk kasir, dan untuk pelanggan. Hasil dari pertanyaan yang telah diberikan ditunjukkan pada lampiran. Pada tabel 7.11 menunjukkan pertanyaan yang akan diajukan kepada tester.

Tabel 7.17 Pertanyaan UAT

No	Pertanyaan	Kriteria	Tester
1	Apakah sistem informasi SFA ini dapat mudah untuk dipahami?	<i>Functional Correctness and Completeness</i>	Pelanggan, Manager, Penjual, dan Kasir
2	Dengan fitur pengelolaan data pegawai apakah memudahkan manager untuk melakukan pengelolaan data pegawai?		Manager
3	Dengan fitur pengelolaan data produk apakah memudahkan manager untuk melakukan pengelolaan data produk?		Manager, Penjual, dan Kasir
4	Dengan fitur melihat data laporan penjualan apakah memudahkan manager untuk memeriksa laporan data penjualan?		Manager
5	Dengan fitur pengelolaan data pelanggan apakah memudahkan manager untuk melakukan pengelolaan data pelanggan?		Manager dan Penjual
6	Dengan fitur pengelolaan data penjualan apakah memudahkan penjual untuk mengelola data penjualan?		Penjual dan Kasir
7	Dengan adanya sistem informasi SFA ini dapat memberikan kepuasan kepada pelanggan?		Pelanggan
8	Apakah pelanggan dapat lebih mudah untuk mengetahui barang yang tersedia di toko?		
9	Apakah dengan adanya sistem SFA dapat meningkatkan pelayanan terhadap pelanggan?	<i>Usability</i>	Manager dan Penjual
10	Apakah informasi yang diberikan kepada pelanggan		Manager, Penjual dan Pelanggan

Tabel 7.17 Pertanyaan UAT(lanjutan)

No	Pertanyaan	Kriteria	Tester
	sudah sesuai dengan kebutuhan pelanggan?		
11	Apakah anda sebagai pelanggan senang dengan tampilan web yang ada saat ini?		Pelanggan
12	Apakah sistem SFA mempermudah dalam penyampaian informasi terkait dengan produk yang ada?	<i>Confidentiality and Availability</i>	Manager dan Penjual
13	Apakah anda sebagai pegawai merasa terjamin dengan keamanan data dari sistem informasi SFA?		Manager, Penjual dan Kasir
14	Apakah anda sebagai pelanggan merasa data yang diberikan kepada toko terjamin keamanannya?		Pelanggan

Setelah didapatkan kriteria dan pertanyaan yang sesuai dengan kriteria yang dibutuhkan. Tahap selanjutnya adalah dengan menganalisa hasil wawancara yang terlampir pada lampiran C hasil dokumen UAT. Pada Tabel 7.12 menunjukan hasil analisa UAT yang telah diberikan kepada tester.

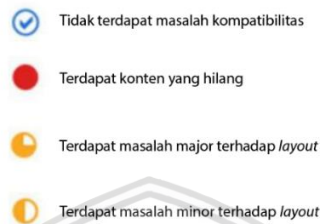
Tabel 7.18 Hasil Wawancara UAT

No	Kriteria	Jumlah Pertanyaan Diajukan	Jawaban "YA"	Jawaban "TIDAK"	Presentase Jawaban "YA"
1	<i>Functional Correctness and Completeness</i>	15	15	0	100%
2	<i>Usability</i>	6	6	0	100%
3	<i>Confidentiality and Availability</i>	6	6	0	100%

Berdasarkan hasil wawancara UAT pada tabel 7.12 menunjukan 100% dari keseluruhan pertanyaan yang diberikan memberikan jawaban "YA". Maka dapat diambil kesimpulan bahwa sistem informasi SFA dapat diterima dengan baik dan sudah sesuai dengan kebutuhan yang disepakati diawal.

7.3 Compatibility Testing

Compatibility testing merupakan sebuah pengujian yang bertujuan untuk mengetahui kemampuan sistem dalam menunjukkan kesesuaian terhadap *web browser* yang digunakan. *Compatibility testing* dilakukan dengan menggunakan software SortSite versi 5.28.810.0. Terdapat beberapa kategori yang ditunjukkan dalam bentuk simbol untuk mengetahui tingkat kompatibilitas yang dijelaskan dalam gambar 7.1.



Gambar 7.1 Simbol Compatibility Testing pada SortSite

Compatibility testing sistem informasi SFA dilakukan pada laman yang diakses oleh pengguna. Hasil *compatibility testing* yang terdapat dalam gambar 7.2 memperlihatkan bahwa terdapat beberapa masalah minor pada bagian *layout* laman situs ketika situs diakses melalui akses aplikasi *web browser* Internet Explorer versi 9 sampai dengan 11, *web browser* Firefox versi 58 dan aplikasi *web browser* BlackBerry dengan versi 7.1 atau sebelumnya.

Browser	IE			Edge	Firefox	Safari		Opera	Chrome	iOS			Android		BlackBerry	
Version	9	10	11	16	58	≤ 10	11	49	63	≤ 9	10	11	≤ 3	4*	≤ 7.1	10.0
Critical Issues	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Major Issues	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Minor Issues	⚠	⚠	⚠	✓	⚠	✓	✓	✓	✓	✓	✓	✓	✓	✓	⚠	✓

Gambar 7.2 Hasil Compatibility Testing pada SortSite

BAB 8 PENUTUP

8.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat diambil kesimpulan sebagai berikut:

1. Hasil analisis kebutuhan pada penelitian ini adalah spesifikasi kebutuhan yang menghasilkan 10 fitur dengan 21 kebutuhan fungsional dan 1 kebutuhan non fungsional yaitu kompatibilitas yang memberikan gambaran umum kemampuan yang dimiliki sistem informasi SFA untuk mengelola data penjualan, mengelola data pegawai, mengelola data pelanggan, dan mengelola data produk. Pada bagian use case menghasilkan informasi terkait tujuan dari penggunaan sistem dan beberapa penjelasan aktivitas yang dilakukan oleh pengguna dan sistem pada saat menggunakan sistem.
2. Pengembangan sistem informasi berbasis web yang telah dibuat dengan menggunakan metode *prototyping* menghasilkan 1 kali perulangan yang didapatkan dengan melakukan demonstrasi sistem dilampirkan pada lampiran B. Hasil perulangan tersebut berupa penambahan fungsi pada fitur melihat pencapaian pegawai untuk melihat secara rinci poin yang didapatkan pegawai dan menambah fungsi pada fitur melihat laporan penjualan untuk dapat melakukan export kedalam excel. Fitur lain yang diberikan melalui sistem informasi yang telah dibangun yaitu, pengelolaan data produk, pengelolaan data penjualan, pengelolaan data pelanggan, pengelolaan data pegawai, melihat rincian bonus pegawai, mencetak laporan penjualan dan memberikan kemudahan bagi pelanggan untuk melihat dan melakukan pembelian secara *online*.
3. Pada pengujian sistem informasi yang telah dibuat didapatkan hasil *validation testing* pada fungsi yang berkaitan dengan pengelolaan data produk, pengelolaan data penjualan, pengelolaan data pegawai dan melakukan pembelian dinyatakan valid. Hasil UAT menunjukkan bahwa sistem dapat diterima 100% dari sisi perusahaan sebagai manager, penjual dan kasir dengan seluruh jawaban "YA" pada kuisisioner yang dibagikan dan menjelaskan bahwa sistem dapat membantu kinerja para pegawai untuk melakukan transaksi penjualan dengan pelanggan yang terlampir pada lampiran C. Selanjutnya, hasil *compatibility testing* terdapat masalah minor pada aplikasi web browser Internet Explorer 9 sampai dengan 11, web browser Firefox versi 58 dan aplikasi web browser BlackBerry dengan versi 7.1 atau sebelumnya.

8.2 Saran

1. Pengembangan sistem lebih lanjut dengan mengembangkan sistem informasi SFA dengan penambahan fitur-fitur yang sesuai dengan proses bisnis dan berkaitan dengan *Sales Force Automation*(SFA) menyesuaikan kebutuhan perusahaan kedepannya. Seperti jika ada pengembalian barang dari pelanggan dapat ditambahkan fitur untuk melakukan retur penjualan.

2. Pada bagian antarmuka perlu dilakukan evaluasi pengguna secara lebih mendalam untuk mengetahui apakah perbaikan perlu dilakukan atau melakukan pengembangan lebih lanjut pada bagian antarmuka pengguna.



DAFTAR PUSTAKA

- Agarwal, B. B., Tayal, S. P. & Gupta, M., 2010. *Software Engineering & Testing*. 1st Edition. Massachusetts: Jones and Bartlett.
- A.S, Rosa, dan M. Shalahudin. 2015. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Booch, G., Rumbaugh, J. & Jacobson, I. 1998. *Unified Modeling Language, The*. 1st Edition. Massachusetts: Addison-Wesley
- Buttle, Francis, 2009. *Customer Relationship Management, concepts and technologies*, 2nd Edition. Elsevier Ltd.
- Dharwiyanti, S., Wahono, R.S., 2003. *Pengantar Unified Modeling Language*. Tersedia melalui: <http://rosnigj.staff.gunadarma.ac.id/Downloads/files/14321/10.+Unified+Modeling+Language.pdf> [Diakses 7 April 2017]
- Gray, Paul & Byun, Jongbok 2001. *Customer Relationship Management*. Center for Research on Information Technology and Organizations. California
- Indriati. 2010. *Pengujian Validasi*. [Online] Tersedia di: <<http://indryz.lecture.ub.ac.id/files/2010/11/Pengujian-Validasi.docx>> [Diakses 25 Oktober 2017].
- Kalakota, R. & Robinson, M., 2001. *E-Business 2.0 Roadmap for Success*. Massachusetts: Addison Wesley Longman
- Naik, K. & Tripathy, P., 2008. *Software Testing And Quality Assurance*. In: *Theory and Practice*. s.l.:s.n., p. 452.
- Object Management Group, 2011. Business Process Model and Notation (BPMN). [pdf] Object Management Group. Tersedia di: <<http://www.omg.org/cgi-bin/doc?formal/13-12-09.pdf>> [Diakses 25 Oktober 2017]
- Payne, Adrian, 2005. *Handbook of CRM: Achieving Excellence in Customer Management*. Elsevier Ltd
- Pressman, R. S., 2010. *Software Engineering: A Practitioner's Approach*. 7th Edition. New York: McGraw-Hill.
- Reid, Allan & Lorenz, Jim, 2008. *Working at a Small-to-Medium Business or ISP, CCNA Discovery Learning Guide*. Longman Group
- Riyanto, 2011. *Membuat Aplikasi E-Commerce dengan PHP & MySQL menggunakan CodeIgniter & JQuery*. Penerbit. CV Andi Offset
- Shainesh, G. & Seth, Jagdish N., 2006. *Customer Relationship Management, a strategic perspective*. Laxmi Publications Pvt. Ltd
- Silver, B., 2012. *BPMN 2.0 Handbook*. 2nd ed. Florida: Future Strategist.

Soliman, Hisham Sayed, 2011. *Customer Relationship and Its Relationship to the Marketing Performance*. Tersedia melalui: http://ijbssnet.com/journals/Vol.%202_No._10;_June_2011/18.pdf [Diakses 8 Maret 2017]

Sommerville, Ian. 2011. *Software Engineering, Ninth Edition*. New York: Addison-Wesley.

Sparx Systems, 2004. *The Business Process Model*. [pdf] Sparx Systems. Tersedia melalui: http://www.sparxsystems.com.au/downloads/whitepapers/The_Business_Process_Model.pdf [Diakses 20 Oktober 2017]

Urdhwarshie, Ashwin, 2016. *Object-Oriented Programming and its Concepts*. International Journal of Innovation and Scientific Research, [online] Tersedia di: <http://www.issr-journals.org/links/papers.php?journal=ijisr&application=pdf&article=IJISR-15-338-03> [Diakses 16 Maret 2017]

Verma, S., 2014. *Analysis of Strengths and Weakness of SDLC Models*. India: Delhi University.

Vilet, Hans van, 2007. *Software Engineering: Principles and Practices*. Wiley Publishing, Inc.

Wong, Tom & Kao, Liz 2007. *Salesforce.com For Dummies, 2nd Edition*. Wiley Publishing, Inc.

